

TRABAJO PRACTICO N°4

# BASE DE DATOS II



**INTEGRANTES COHORTE 2024:** BENITEZ DARIO SEBASTIAN,  
FACHINOTTI PABLO DARIO, GOMEZ JOSE HIPOLITO, HERRERA  
VALENTIN, LOPEZ ROBERTO JOSE, NOBLEGA VICTOR, NUÑEZ  
GALLARDO GIMENA SOLEDAD

**ACTIVIDAD:**

- 1- Realizar cuestionario para asistencia en campus
- 2-Trabajo Grupal: Aplicar sentencias vistas en clase , enviar capturas de lo realizado con nombre y apellido de quienes trabajaron. Fecha de entrega 13/05

**AÑADIENDO ÍNDICES EN MONGO DB:**

```
test> use paso
      switched to db paso
```

```
paso> db.paso.insertMany([{"nombre": "Roberto", "apellido": "Lopez"}, {"nombre": "Lucia", "apellido": "Ramirez"}, {"nombre": "carlos", "apellido": "martinez"}])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6840c0276dd3ee137eb5f899'),
    '1': ObjectId('6840c0276dd3ee137eb5f89a'),
    '2': ObjectId('6840c0276dd3ee137eb5f89b')
  }
}
paso> db.paso.getIndexes()
[ { v: 2, key: { _id: 1 }, name: '_id_' } ]
```

***Búsqueda de datos más lenta***

```
db.paso.find({"nombre": "roberto"})
paso> db.paso.find({"nombre": "Roberto"})
[
  {
    "_id": ObjectId('6840c0276dd3ee137eb5f899'),
    "nombre": 'Roberto',
    apellido: 'Lopez'
  }
]
paso> |
```

***Búsqueda más eficaz***

```
db.paso.find({ "_id": ObjectId("xxxxxxxx") })
```

```
paso> db.paso.find({ "_id": ObjectId("6840c0276dd3ee137eb5f899") })
[  
  {  
    _id: ObjectId('6840c0276dd3ee137eb5f899'),  
    'nombre': 'Roberto ',  
    apellido: 'Lopez'  
  }  
]
```

db.paso.createIndex({"apellido": 1})

👉 ¿Qué hace esta sentencia?

Crea un **índice** sobre el campo apellido para que las búsquedas como:  
db.paso.find({apellido: "Lopez"})  
sean más rápidas y eficientes.

```
paso> db.paso.createIndex({"apellido": 1})
apellido_1
paso> db.paso.find({apellido: "Lopez"})
[  
  {  
    _id: ObjectId('6840c0276dd3ee137eb5f899'),  
    'nombre': 'Roberto ',  
    apellido: 'Lopez'  
  }  
]
paso> |
```

**Para crear un index:**

```
paso> db.paso2.insertMany([
...   {"usuario": "Fer_4T", "contraseña": "12345d", "correo": "fer@ejemplo.com"},  
...   {"usuario": "maria", "contraseña": "12345f", "correo": "maria@ejemplo.com"},  
...   {"usuario": "kev", "contraseña": "12345k", "correo": "kev@ejemplo.com"}  
... ])
...
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6840c4096dd3ee137eb5f89c'),
    '1': ObjectId('6840c4096dd3ee137eb5f89d'),
    '2': ObjectId('6840c4096dd3ee137eb5f89e')
  }
}
```

**Nos mostrará la colección de datos creada.**

```
paso> db.paso2.find()
[
  {
    _id: ObjectId('6840c4096dd3ee137eb5f89c'),
    usuario: 'Fer_4T',
    'contraseña': '12345d',
    correo: 'fer@ejemplo.com'
  },
  {
    _id: ObjectId('6840c4096dd3ee137eb5f89d'),
    usuario: 'maria',
    'contraseña': '12345f',
    correo: 'maria@ejemplo.com'
  },
  {
    _id: ObjectId('6840c4096dd3ee137eb5f89e'),
    usuario: 'kev',
    'contraseña': '12345k',
    correo: 'kev@ejemplo.com'
  }
]
paso> |
```

```
paso> db.paso2.getIndexes()
[ { v: 2, key: { _id: 1 }, name: '_id_' } ]
```

**PARA CREAR LOS ÍNDICES ÚNICOS**  
**UTILIZAREMOS CREATEINDEX()**

```
paso> db.paso2.createIndex({correo:1},{unique:true})
correo_1
paso> |
```

**PARA CREAR UN ÍNDICE EN SEGUNDO PLANO:**

```
test> use segundo
switched to db segundo
segundo> |
```

```
segundo> db.ejemplo.insertMany([
...   { "nombre": "lautaro", "edad": 15, "fecha": new Date() },
...   { "nombre": "lautaro", "edad": 15, "fecha": new Date() },
...   { "nombre": "gabriela", "edad": 16, "fecha": new Date() },
...   { "nombre": "natalia", "edad": 19, "fecha": new Date() },
...   { "nombre": "david", "edad": 20, "fecha": new Date() }
... ])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6842016eb2bc56b12db5f899'),
    '1': ObjectId('6842016eb2bc56b12db5f89a'),
    '2': ObjectId('6842016eb2bc56b12db5f89b'),
    '3': ObjectId('6842016eb2bc56b12db5f89c'),
    '4': ObjectId('6842016eb2bc56b12db5f89d')
  }
}
segundo> |
```

>db.ejemplo.find().pretty()

Nos mostrará los registros que creamos

Para agregar los índices utilizaremos los nombres y fechas que son índices que podemos agregar.

>db.ejemplo.createIndex({"nombre":1},{"background":true})

>db.ejemplo.createIndex({"fecha":1},{"background":true})

```
segundo> db.ejemplo.find().pretty()
[
  {
    _id: ObjectId('6842016eb2bc56b12db5f899'),
    nombre: 'lautaro',
    edad: 15,
    fecha: ISODate('2025-06-05T20:43:26.955Z')
  },
  {
    _id: ObjectId('6842016eb2bc56b12db5f89a'),
    nombre: 'lautaro',
    edad: 15,
    fecha: ISODate('2025-06-05T20:43:26.955Z')
  },
  {
    _id: ObjectId('6842016eb2bc56b12db5f89b'),
    nombre: 'gabriela',
    edad: 16,
    fecha: ISODate('2025-06-05T20:43:26.955Z')
  },
  {
    _id: ObjectId('6842016eb2bc56b12db5f89c'),
    nombre: 'natalia',
    edad: 19,
    fecha: ISODate('2025-06-05T20:43:26.955Z')
  },
  {
    _id: ObjectId('6842016eb2bc56b12db5f89d'),
    nombre: 'david',
    edad: 20,
    fecha: ISODate('2025-06-05T20:43:26.955Z')
  }
]
segundo> |
```

```
segundo> db.ejemplo.createIndex({"nombre":1}, {"background":true})
nombre_1
segundo> db.ejemplo.createIndex({"fecha":1}, {"background":true})
fecha_1
segundo> |
```

## **"BACKGROUND"**

Esta propiedad le va a dar la orden a Mongo para que éste índice se ejecute los datos en 2do. Plano

- Unas vez creados los índices aplicaremos las siguientes sentencias.

- <db.ejemplo.getIndexes();

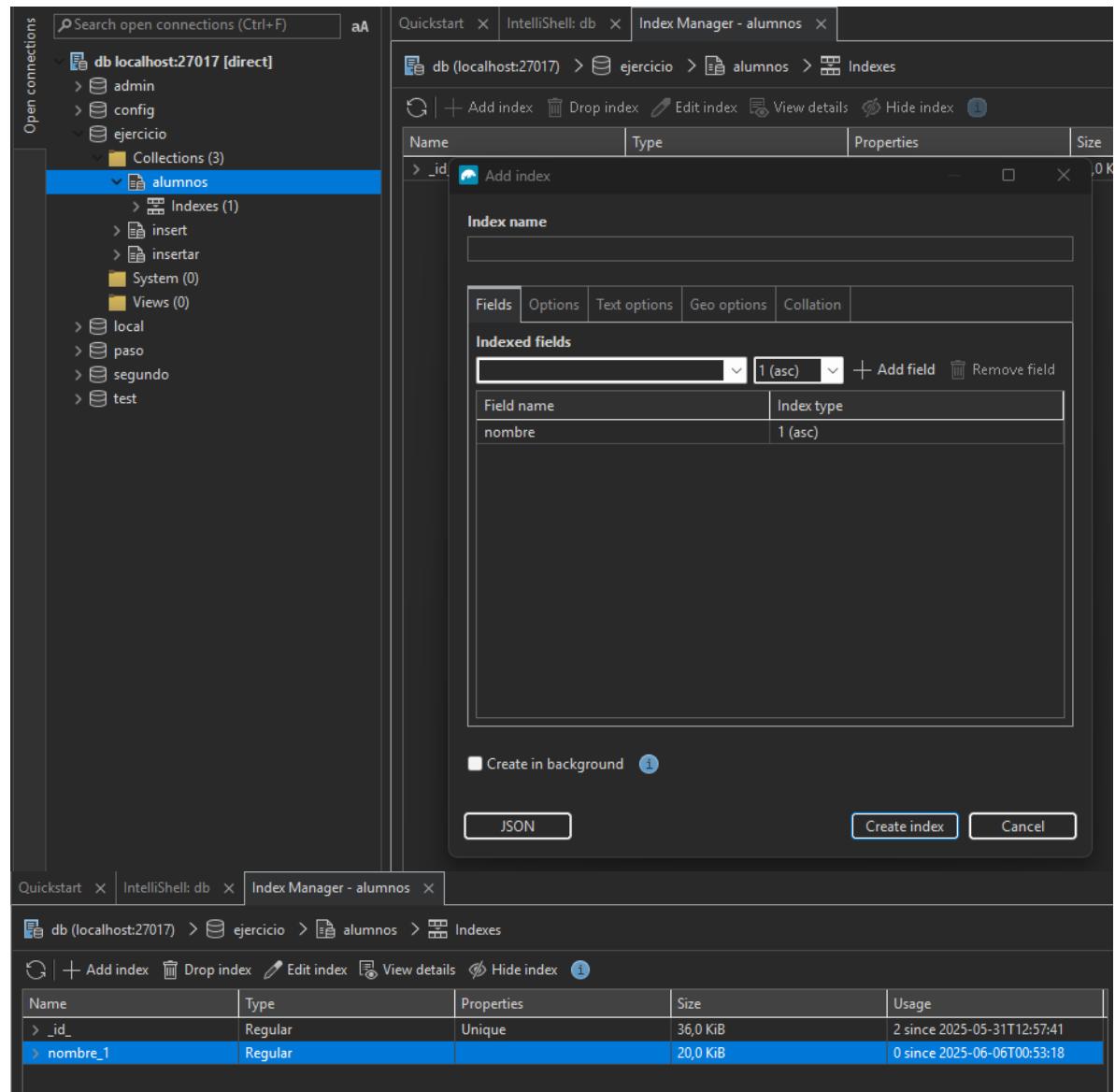
Podremos visualizar los índices creados por defecto y los que cree de forma específica.

Esto permite que no se interrumpa la ejecución de nuestra base de datos mientras otros usuarios estén trabajando en ella.

Mayormente lo utilizan las empresas para agilizar datos

```
segundo> db.ejemplo.getIndexes()
[
  { v: 2, key: { _id: 1 }, name: '_id_' },
  { v: 2, key: { nombre: 1 }, name: 'nombre_1', background: true },
  { v: 2, key: { fecha: 1 }, name: 'fecha_1', background: true }
]
segundo> |
```

## GENERAR INDICES EN STUDIO 3T



The screenshot shows the Studio 3T interface for managing MongoDB databases. On the left, the 'Open connections' sidebar lists 'db localhost:27017 [direct]' with collections: admin, config, ejercicio, local, paso, segundo, and test. Under 'ejercicio', the 'alumnos' collection is selected, showing its sub-collections: insert, insertar, System (0), and Views (0). The main window title is 'Index Manager - alumnos'. The 'Indexes' tab is active, showing two existing indexes: '\_id' (Unique, Regular) and 'nombre\_1' (Regular). A new index is being created for the 'nombre' field, with the index name set to 'nombre'. The 'Fields' tab shows the field 'nombre' with index type '1 (asc)'. Buttons at the bottom include 'Create in background', 'JSON', 'Create index' (highlighted in blue), and 'Cancel'.

Name	Type	Properties	Size	Usage
_id	Regular	Unique	36,0 kB	2 since 2025-05-31T12:57:41
nombre_1	Regular		20,0 kB	0 since 2025-06-06T00:53:18