

UNIDAD 8

BASES DE DATOS NOSQL: MONGODB

BASES DE DATOS 22/23
CFGS DAW

BOLETÍN EJERCICIOS 8.1. MONGODB: DDL Y DQL

Revisado por:

Abelardo Martínez y Pau Miñana

Autor:

Sergio Badal

Licencia Creative Commons



Reconocimiento - NoComercial - CompartirIgual (by-nc-sa): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

ÍNDICE DE CONTENIDO

1. EJERCICIO LIBROS.....	3
2. EJERCICIO ARTÍCULOS.....	5
3. EJERCICIO MEDICAMENTOS.....	7

UD8.1. BOLETÍN MONGODB DDL Y DQL

1. EJERCICIO LIBROS

```
use pruebas
db.libros.drop()
db.libros.insertOne(
{
  _id: 1,
  titulo: 'El aleph',
  autor: 'Borges',
  editorial: 'Planeta',
  cantidad: 50
}
)
db.libros.insertOne(
{
  _id: 2,
  titulo: 'Martin Fierro',
  autor: 'Jose Hernandez',
  editorial: 'Planeta',
  precio: 52,
  cantidad: 12
}
)
db.libros.insertOne(
{
  _id: 3,
  titulo: 'Aprenda PHP',
  autor: 'Mario Molina',
  editorial: 'Siglo XXI',
  precio: 50,
  cantidad: 20
}
)
db.libros.insertOne(
{
  _id: 4,
  titulo: 'Java en 10 minutos',
  editorial: 'Siglo XXI',
  precio: 45,
  cantidad: 1
})
```

10. Recuperar todos los libros que en el campo cantidad tienen 50 o más:

```
db.libros.find( { cantidad: { $gte : 50 } }).pretty()
```

11. Recuperar toda la información sobre el libro más barato, incluyendo el id

12. Recuperar el título e id de los libros que NO tienen un precio indicado:

13. Recuperar todos los libros que en el campo cantidad hay un valor distinto a 50:

14. Recuperar todos los libros cuyo precio esté comprendido entre 20 y 45:

15. Recuperar todos los libros de la editorial 'Planeta':

16. Recuperar todos los libros que no pertenezcan a la editorial 'Planeta':

17. Recuperar los libros que tienen un precio mayor o igual a 50 o la cantidad es 1

18. Recuperar solo el título de los tres primeros libros con precio mayor que 20, ordenados por precio descendente, omitiendo el id

19. Recuperar el título y precio del libro más caro, omitiendo el id

2. EJERCICIO ARTÍCULOS

```
use pruebas
db.articulos.drop()

db.articulos.insertOne(
{
  _id: 1,
  nombre: 'MULTIFUNCION HP DESKJET 2675',
  rubro: 'impresora',
  precio: 3000,
  stock: 20
}
)
db.articulos.insertOne(
{
  _id: 2,
  nombre: 'MULTIFUNCION EPSON EXPRESSION XP241',
  rubro: 'impresora',
  precio: 3700,
  stock: 5
}
)
db.articulos.insertOne(
{
  _id: 3,
  nombre: 'LED 19 PHILIPS',
  rubro: 'monitor',
  precio: 4500,
  stock: 2
}
)
db.articulos.insertOne(
{
  _id: 4,
  nombre: 'LED 22 PHILIPS',
  rubro: 'monitor',
  precio: 5700,
  stock: 4
}
)
```

```
db.articulos.insertOne(  
  {  
    _id: 5,  
    nombre: 'LED 27 PHILIPS',  
    rubro: 'monitor',  
    precio: 12000,  
    stock: 1  
  }  
)
```

```
db.articulos.insertOne(  
  {  
    _id: 6,  
    nombre: 'LOGITECH M90',  
    rubro: 'mouse',  
    precio: 300,  
    stock: 4  
  }  
)
```

20. Imprimir todos los documentos de la colección 'articulos' que no son impresoras

```
db.articulos.find({ rubro : { $ne: 'impresora' } }).pretty()
```

21. Imprimir todos los artículos que pertenecen al rubro de 'mouse'

22. Imprimir todos los artículos con un precio mayor o igual a 5000

23. Imprimir todas las impresoras que tienen un precio mayor o igual a 3500

24. Imprimir todos los artículos cuyo stock se encuentra comprendido entre 0 y 4

25. Imprimir el nombre de los artículos "monitor o impresora", ordenados por precio descendente

3. EJERCICIO MEDICAMENTOS

```
use pruebas
db.medicamentos.drop()

db.medicamentos.insertOne(
{
  _id: 1,
  nombre: 'Sertal',
  laboratorio: 'Roche',
  precio: 5.2,
  cantidad: 101
}
)
db.medicamentos.insertOne(
{
  _id: 2,
  nombre: 'Buscapina',
  laboratorio: 'Roche',
  precio: 4.10,
  cantidad: 200
}
)
db.medicamentos.insertOne(
{
  _id: 3,
  nombre: 'Amoxidal 500',
  laboratorio: 'Bayer',
  precio: 15.60,
  cantidad: 140
}
)
db.medicamentos.insertOne(
{
  _id: 4,
  nombre: 'Paracetamol 500',
  laboratorio: 'Bago',
  precio: 1.90,
  cantidad: 300
}
)
db.medicamentos.insertOne(
{
```

```
_id: 5,
nombre: 'Bayaspirina',
laboratorio: 'Bayer',
precio: 2.10,
cantidad: 100
}
)
db.medicamentos.insertOne(
{
  _id: 6,
  nombre: 'Amoxidal jarabe',
  laboratorio: 'Bayer',
  precio: 5.10,
  cantidad: 50
}
)
```

30. Recupera los medicamentos cuyo laboratorio sea 'Roche' y cuyo precio sea menor a 5

```
db.medicamentos.find({laboratorio:'Roche',precio:{<5}}).pretty()
```

31. Muestra todos los medicamentos cuyo laboratorio NO sea "Bayer"

32. Muestra todos los medicamentos cuyo laboratorio sea "Bayer" y cuya cantidad NO sea=100

33. Recupera el nombre y laboratorio de los medicamentos con mayor stock (por cantidad descendente)

34. Recupera el nombre y el laboratorio del medicamento con menor stock

35. Recupera el nombre e id de los dos medicamentos con mayor stock, ordenados por stock desc

36. Recupera todos los datos del medicamento más barato, excepto el id