



BASES DE DATOS PARA COMPUTO EN LA NUBE

**Administración de Bases de Datos NoSQ:
operaciones CRUD**

UNIVERSIDAD TECNOLOGICA DE CANCUN

TSU. Desarrollo Software Multiplataforma

ANGEL MANUEL ROSALDO ADAME

22393273

SM54

19/01/2024

EJERCICIOS OPERACIONES CRUD

1-Crear la colección 'articulos' en la base de datos 'productos', cargar luego 6 documentos:

2-Mostrar todos los documentos de la colección 'articulos':

```
productos> db.articulos.find()
[
  {
    _id: 1,
    nombre: 'MULTIFUNCION HP DESKJET 2675',
    rubro: 'impresora',
    precio: 3000,
    stock: 20
  },
  {
    _id: 2,
    nombre: 'MULTIFUNCION EPSON EXPRESSION XP241',
    rubro: 'impresora',
    precio: 3700,
    stock: 5
  },
  {
    _id: 3,
    nombre: 'LED 19 PHILIPS',
    rubro: 'monitor',
    precio: 4500,
    stock: 2
  },
  {
    _id: 4,
    nombre: 'LED 22 PHILIPS',
    rubro: 'monitor',
    precio: 5700,
    stock: 4
  },
  {
    _id: 5,
    nombre: 'LED 27 PHILIPS',
    rubro: 'monitor',
    stock: 1
  },
  {
    _id: 6,
    nombre: 'LOGITECH M90',
    rubro: 'mouse',
    precio: 300,
    stock: 4
  }
]
```

3- Mostrar todos los documentos de la colección 'articulos' que no son impresoras:

```
productos> db.articulos.find({ rubro: { $ne: "impresora" } })
[
  {
    _id: 3,
    nombre: 'LED 19 PHILIPS',
    rubro: 'monitor',
    precio: 4500,
    stock: 2
  },
  {
    _id: 4,
    nombre: 'LED 22 PHILIPS',
    rubro: 'monitor',
    precio: 5700,
    stock: 4
  },
  {
    _id: 5,
    nombre: 'LED 27 PHILIPS',
    rubro: 'monitor',
    precio: 12000,
    stock: 1
  },
  {
    _id: 6,
    nombre: 'LOGITECH M90',
    rubro: 'mouse',
    precio: 300,
    stock: 4
  }
]
```

4-Mostrar todos los artículos que pertenecen al rubro de 'mouse':

```
productos> db.articulos.find({rubro: "mouse"})
[
  {
    _id: 6,
    nombre: 'LOGITECH M90',
    rubro: 'mouse',
    precio: 300,
    stock: 4
  }
]
productos>
```

5- Mostrar todos los artículos con un precio mayor o igual a 5000:

```
productos> db.articulos.find({ "precio": { "$gte": 5000 } })
[
  {
    _id: 4,
    nombre: 'LED 22 PHILIPS',
    rubro: 'monitor',
    precio: 5700,
    stock: 4
  },
  {
    _id: 5,
    nombre: 'LED 27 PHILIPS',
    rubro: 'monitor',
    precio: 12000,
    stock: 1
  }
]
```

6- Mostrar todas las impresoras que tienen un precio mayor o igual a 3500:

```
productos> db.articulos.find({ rubro: "impresora", precio: { $gte: 3500 } })
[
  {
    _id: 2,
    nombre: 'MULTIFUNCION EPSON EXPRESSION XP241',
    rubro: 'impresora',
    precio: 3700,
    stock: 5
  }
]
```

7- Mostrar todos los artículos cuyo stock se encuentra comprendido entre 0 y 4:

```
productos> db.articulos.find({ stock: { $gte: 0, $lte: 4 } })
[
  {
    _id: 3,
    nombre: 'LED 19 PHILIPS',
    rubro: 'monitor',
    precio: 4500,
    stock: 2
  },
  {
    _id: 4,
    nombre: 'LED 22 PHILIPS',
    rubro: 'monitor',
    precio: 5700,
    stock: 4
  },
  {
    _id: 5,
    nombre: 'LED 27 PHILIPS',
    rubro: 'monitor',
    precio: 12000,
    stock: 1
  },
  {
    _id: 6,
    nombre: 'LOGITECH M90',
    rubro: 'mouse',
    precio: 300,
    stock: 4
  }
]
```

2da parte Ejercicios propuestos:

1-Retoma la colección 'articulos' en la base de datos 'productos':

```
switched to db productos
productos> show collections
articulos
clientes
libros
```

2- Mostrar todos los documentos de la colección 'articulos':

```
productos> db.articulos.find()
[
  {
    _id: 1,
    nombre: 'MULTIFUNCION HP DESKJET 2675',
    rubro: 'impresora',
    precio: 3000,
    stock: 20
  },
  {
    _id: 2,
    nombre: 'MULTIFUNCION EPSON EXPRESSION XP241',
    rubro: 'impresora',
    precio: 3700,
    stock: 5
  },
  {
    _id: 3,
    nombre: 'LED 19 PHILIPS',
    rubro: 'monitor',
    precio: 4500,
    stock: 2
  },
  {
    _id: 4,
    nombre: 'LED 22 PHILIPS',
    rubro: 'monitor',
    precio: 5700,
    stock: 4
  },
  {
    _id: 5,
    nombre: 'LED 27 PHILIPS',
    rubro: 'monitor',
    precio: 12000,
    stock: 1
  },
  {
    _id: 6,
    nombre: 'LOGITECH M90',
    rubro: 'mouse',
    precio: 300,
    stock: 4
  }
]
```

3- Modificar el precio del mouse 'LOGITECH M90' a 110.00:

```
productos> db.articulos.updateOne(
...   { nombre: "LOGITECH M90", rubro: "mouse" }, { $set: { precio: 110.00 } })
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

```
{
  _id: 6,
  nombre: 'LOGITECH M90',
  rubro: 'mouse',
  precio: 110,
  stock: 0
}
```

4- Fijar el stock en 0 del artículo cuyo _id es 6:

```
productos> db.articulos.updateOne( { _id: 6 }, { $set: { stock: 0 } })
{ acknowledged: true,
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,,
  modifiedCount: 0,
  upsertedCount: 0
}
```

```
{
  _id: 6,
  nombre: 'LOGITECH M90',
  rubro: 'mouse',
  precio: 110,
  stock: 0
}
```

5-Agregar el campo proveedores con el array ['Martinez','Gutierrez'] para el artículo cuyo _id es 6:

```
productos> db.articulos.updateOne(
...   { _id: 6 }, { $set: { proveedores: ['Martinez', 'Gutierrez'] } })
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

```

{
  _id: 6,
  nombre: 'LOGITECH M90',
  rubro: 'mouse',
  precio: 110,
  stock: 0,
  proveedores: [ 'Martinez', 'Gutierrez' ]
}
]

```

6-Eliminar el campo proveedores para el artículo cuyo _id es 6:

```

productos> db.articulos.updateOne(
...   { _id: 6 }, { $unset: { proveedores: "" } })
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

```

{
  _id: 6,
  nombre: 'LOGITECH M90',
  rubro: 'mouse',
  precio: 110,
  stock: 0
}

```

3ra parte Ejercicios propuestos

1.- Crear la colección 'libros' en la base de datos 'biblioteca' (eliminar la colección previamente), cargar luego 4 documentos:

```
biblioteca> db.libros.insertOne(
... {
...   _id: 4,
...   titulo: 'Java en 10 minutos',
...   autor: {
...     nombre: 'Mario Molina',
...     nacionalidad: 'Española'
...   },
...   editorial: ['Siglo XXI'],
...   precio: 45,
...   cantidad: 1
... }
... );
{ acknowledged: true, insertedId: 4 }
biblioteca>
```

2-Mostrar todos los documentos de la colección 'libros':

```
biblioteca> db.libros.find()
[
  {
    _id: 1,
    titulo: 'El aleph',
    autor: { nombre: 'Borges', nacionalidad: 'Argentina' },
    editorial: [ 'Siglo XXI', 'Planeta' ],
    precio: 20,
    cantidad: 50
  },
  {
    _id: 2,
    titulo: 'Martin Fierro',
    autor: { nombre: 'Jose Hernandez', nacionalidad: 'Argentina' },
    editorial: [ 'Siglo XXI' ],
    precio: 50,
    cantidad: 12
  },
  {
    _id: 3,
    titulo: 'Aprenda PHP',
    autor: { nombre: 'Mario Molina', nacionalidad: 'Española' },
    editorial: [ 'Siglo XXI', 'Planeta' ],
    precio: 50,
    cantidad: 20
  },
  {
    _id: 4,
    titulo: 'Java en 10 minutos',
    autor: { nombre: 'Mario Molina', nacionalidad: 'Española' },
    editorial: [ 'Siglo XXI' ],
    precio: 45,
    cantidad: 1
  }
]
```


3-Mostrar todos los libros de autores de nacionalidad 'Argentina':

```
biblioteca> db.libros.find({ "autor.nacionalidad": "Argentina" })
[
  {
    _id: 1,
    titulo: 'El aleph',
    autor: { nombre: 'Borges', nacionalidad: 'Argentina' },
    editorial: [ 'Siglo XXI', 'Planeta' ],
    precio: 20,
    cantidad: 50
  },
  {
    _id: 2,
    titulo: 'Martin Fierro',
    autor: { nombre: 'Jose Hernandez', nacionalidad: 'Argentina' },
    editorial: [ 'Siglo XXI' ],
    precio: 50,
    cantidad: 12
  }
]
```

4-Mostrar los libros de 'Borges':

```
biblioteca> db.libros.find({ "autor.nombre": "Borges" })
[
  {
    _id: 1,
    titulo: 'El aleph',
    autor: { nombre: 'Borges', nacionalidad: 'Argentina' },
    editorial: [ 'Siglo XXI', 'Planeta' ],
    precio: 20,
    cantidad: 50
  }
]
```

5-Mostrar todos los libros de nacionalidad 'Española' que cuestan 50 o más:

```
biblioteca> db.libros.find({
...   "autor.nacionalidad": "Española",
...   precio: { $gte: 50 }
... })
[
  {
    _id: 3,
    titulo: 'Aprenda PHP',
    autor: { nombre: 'Mario Molina', nacionalidad: 'Española' },
    editorial: [ 'Siglo XXI', 'Planeta' ],
    precio: 50,
    cantidad: 20
  }
]
```

6- Crear la colección 'series' en la base de datos 'programas', cargar luego 2 documentos:

```
... temporada2: [
... {
...   capitulo1:{
...     titulo:'Lo que queda por delante',
...     audiencia:12000000
...   }
... },
... {
...   capitulo2:{
...     titulo:'Sangría',
...     audiencia:13000000
...   }
... }
... ]
... }
... );
{ acknowledged: true, insertedId: 2 }
```

7- Mostrar todos los documentos de la colección 'series'.

```
programas> db.series.find()
[
  {
    _id: 1,
    titulo: 'The big bang theory',
    productor: 'Chuck Lorre',
    actores: [
      'Johnny Galecki',
      'Jim Parsons',
      'Kaley Cuoco',
      'Kunal Nayyar',
      'Simon Helberg',
      'Mayim Bialik',
      'Melissa Rauch'
    ],
    temporada1: [
      { capitulo1: { titulo: 'Piloto', audiencia: 8300000 } },
      {
        capitulo2: { titulo: 'La hipótesis del Gran Cerebro', audiencia: 8700000 }
      },
      {
        capitulo3: {
          titulo: 'El Corolario de el Gato con Botas',
          audiencia: 9200000
        }
      }
    ],
    temporada2: [
      {
        capitulo1: {
          titulo: 'El paradigma del pescado malo',
          audiencia: 10000000
        }
      },
      {
        capitulo2: { titulo: 'La topología de la bragueta', audiencia: 11000000 }
      }
    ]
  },
  {
    _id: 2,
    titulo: 'The Walking Dead',
    productor: 'Robert Kirkman',
```

8-Mostrar los datos de todas las temporadas de la serie 'The big bang theory':

```
programas> db.series.find({ titulo: 'The big bang theory' })
[
  {
    _id: 1,
    titulo: 'The big bang theory',
    productor: 'Chuck Lorre',
    actores: [
      'Johnny Galecki',
      'Jim Parsons',
      'Kaley Cuoco',
      'Kunal Nayyar',
      'Simon Helberg',
      'Mayim Bialik',
      'Melissa Rauch'
    ],
    temporada1: [
      { capitulo1: { titulo: 'Piloto', audiencia: 8300000 } },
      {
        capitulo2: { titulo: 'La hipótesis del Gran Cerebro', audiencia: 8700000 }
      },
      {
        capitulo3: {
          titulo: 'El Corolario de el Gato con Botas',
          audiencia: 9200000
        }
      }
    ],
    temporada2: [
      {
        capitulo1: {
          titulo: 'El paradigma del pescado malo',
          audiencia: 10000000
        }
      },
      {
        capitulo2: { titulo: 'La topología de la bragueta', audiencia: 11000000 }
      }
    ]
  }
]
```

9- Mostrar los datos de todos los capítulos de la primera temporada de la serie de 'The big bang theory':

```
programas> db.series.find({ titulo: 'The Big Bang theory' }, { temporada1: 1, _id: 0 })
[
  {
    temporada1: [
      { capitulo1: { titulo: 'Piloto', audiencia: 8300000 } },
      {
        capitulo2: { titulo: 'La hipótesis del Gran Cerebro', audiencia: 8700000 }
      },
      {
        capitulo3: {
          titulo: 'El Corolario de el Gato con Botas',
          audiencia: 9200000
        }
      }
    ]
  }
]
```

10- Mostrar solo el primer capítulo de la primera temporada de 'The Walking Dead':

```
programas> db.series.find({ titulo: 'The Walking Dead' }, { "temporada1.capitulo1": 1, _id: 0 })
[
  {
    temporada1: [ { capitulo1: { titulo: 'TS 19', audiencia: 7000000 } }, {}, {} ]
  }
]
```