**Aprende haciendo**

**Taller: Consultas (Orden y Límites)**

Con este taller usaremos las instrucciones en MongoDB para hacer consultas más avanzadas con relación al orden y límites.

Implementaremos en MongoDB una base de datos llamada **empresa** que incluye una colección de **proveedores**. Nos han solicitado que hagamos algunas consultas a los datos contenidos en dicha base de datos.

Realiza lo siguiente:

1. Crea la base de datos **empresa**.

> use empresa

switched to db empresa

>

1. Crea la colección **proveedores**.
2. Agrega cinco documentos:
   * { nombre: “P1”, capital: 100000}
   * { nombre: “P2”, capital: 10000}
   * { nombre: “P3”, capital: 10}
   * { nombre: “P5”, capital: 100}
   * { nombre: “P6”, capital: 1000}

> db.proveedores.insert([{ "nombre" : "P1", "capital" : 100000 },{ "nombre" : "P2", "capital" : 10000 }])

BulkWriteResult({

"writeErrors" : [ ],

"writeConcernErrors" : [ ],

"nInserted" : 2,

"nUpserted" : 0,

"nMatched" : 0,

"nModified" : 0,

"nRemoved" : 0,

"upserted" : [ ]

})

> db.proveedores.insert([{ "nombre" : "P3", "capital" : 10 },{ "nombre" : "P5", "capital" : 100 }])

BulkWriteResult({

"writeErrors" : [ ],

"writeConcernErrors" : [ ],

"nInserted" : 2,

"nUpserted" : 0,

"nMatched" : 0,

"nModified" : 0,

"nRemoved" : 0,

"upserted" : [ ]

})

> db.proveedores.insert([{ "nombre" : "P6", "capital" : 1000 }])

BulkWriteResult({

"writeErrors" : [ ],

"writeConcernErrors" : [ ],

"nInserted" : 1,

"nUpserted" : 0,

"nMatched" : 0,

"nModified" : 0,

"nRemoved" : 0,

"upserted" : [ ]

})

>

1. Realiza una consulta a la colección proveedores cuya salida se muestre en un formato legible usando pretty().

> db.proveedores.find().pretty()

{

"\_id" : ObjectId("60abe6fc47619a0332ee4b52"),

"nombre" : "P1",

"capital" : 100000

}

{

"\_id" : ObjectId("60abe6fc47619a0332ee4b53"),

"nombre" : "P2",

"capital" : 10000

}

{

"\_id" : ObjectId("60abe71947619a0332ee4b54"),

"nombre" : "P3",

"capital" : 10

}

{

"\_id" : ObjectId("60abe71947619a0332ee4b55"),

"nombre" : "P5",

"capital" : 100

}

{

"\_id" : ObjectId("60abe72d47619a0332ee4b56"),

"nombre" : "P6",

"capital" : 1000

}

>

1. Realiza una consulta a la colección proveedores ordenados por el atributo “capital” de forma **ascendente**.

> db.proveedores.find().sort({ "capital" : 1 })

{ "\_id" : ObjectId("60abe71947619a0332ee4b54"), "nombre" : "P3", "capital" : 10 }

{ "\_id" : ObjectId("60abe71947619a0332ee4b55"), "nombre" : "P5", "capital" : 100 }

{ "\_id" : ObjectId("60abe72d47619a0332ee4b56"), "nombre" : "P6", "capital" : 1000 }

{ "\_id" : ObjectId("60abe6fc47619a0332ee4b53"), "nombre" : "P2", "capital" : 10000 }

{ "\_id" : ObjectId("60abe6fc47619a0332ee4b52"), "nombre" : "P1", "capital" : 100000 }

>

1. Realiza una consulta a la colección proveedores ordenados por el atributo “capital” de forma **descendente**.

> db.proveedores.find().sort({ "capital" : -1 })

{ "\_id" : ObjectId("60abe6fc47619a0332ee4b52"), "nombre" : "P1", "capital" : 100000 }

{ "\_id" : ObjectId("60abe6fc47619a0332ee4b53"), "nombre" : "P2", "capital" : 10000 }

{ "\_id" : ObjectId("60abe72d47619a0332ee4b56"), "nombre" : "P6", "capital" : 1000 }

{ "\_id" : ObjectId("60abe71947619a0332ee4b55"), "nombre" : "P5", "capital" : 100 }

{ "\_id" : ObjectId("60abe71947619a0332ee4b54"), "nombre" : "P3", "capital" : 10 }

>

1. Realiza una consulta a la colección proveedores que devuelva los 3 proveedores con mayor capital usando limit().

> db.proveedores.find().sort({ "capital" : -1 }).limit(3)

{ "\_id" : ObjectId("60abe6fc47619a0332ee4b52"), "nombre" : "P1", "capital" : 100000 }

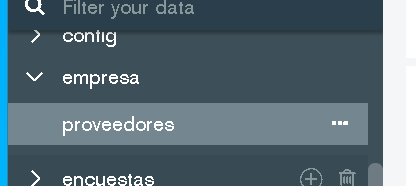
{ "\_id" : ObjectId("60abe6fc47619a0332ee4b53"), "nombre" : "P2", "capital" : 10000 }

{ "\_id" : ObjectId("60abe72d47619a0332ee4b56"), "nombre" : "P6", "capital" : 1000 }

>

**Puntos de Verificación**

1. Verifica que tengas la base de datos creada.
2. Verifica que tengas la colección creada.



1. Realiza un find() de la colección proveedores y verifica que obtienes 5 documentos.
2. Verifica que el comando pretty() funciona viendo la salida que muestran los documentos con un mejor formato.
3. Observa los documentos ordenados de acuerdo con el atributo “capital” en el siguiente orden:
   * { nombre: “P3”, capital: 10}
   * { nombre: “P5”, capital: 100}
   * { nombre: “P6”, capital: 1000}
   * { nombre: “P2”, capital: 10000}
   * { nombre: “P1”, capital: 100000}
4. Observa los documentos ordenados de acuerdo al atributo “capital” en el siguiente orden:
   * { nombre: “P1”, capital: 100000}
   * { nombre: “P2”, capital: 10000}
   * { nombre: “P6”, capital: 1000}
   * { nombre: “P5”, capital: 100}
   * { nombre: “P3”, capital: 10}
5. Observa los documentos ordenados de acuerdo con el atributo “capital” en el siguiente orden:
   * { nombre: “P1”, capital: 100000}
   * { nombre: “P2”, capital: 10000}
   * { nombre: “P6”, capital: 1000}

Esperamos que ahora te sientas más seguro(a) realizando las funciones de creación e inserción en MongoDB y las correspondientes operaciones de consultas.