

# **Proyecto Trimestral 2**

**Aggregate, Mongoimport, Mongoexport y Mongo Atlas.**

*-Gestión de Bases de Datos NoSQL-*

## Índice de contenido

|  |    |
|--|----|
| 1.Estructuras de colecciones.....                        | 3  |
| 1.1.Colección soldItems.....                             | 4  |
| 1.2.Colección sales.....                                 | 6  |
| 1.3.Colección providers.....                             | 6  |
| 1.4.Colección products.....                              | 8  |
| 2.Uso de Mongo Atlas.....                                | 13 |
| 2.1.Conectar una base de datos usando mongo shell.....   | 16 |
| 2.2.Conectar una base de datos usando mongo compass..... | 16 |
| 3.Creación de la base de datos.....                      | 17 |
| 3.1.Inserts.....   | 17 |
| 3.2.Outs.....  | 19 |
| 3.3.Queries.....   | 27 |
| 4.Uso de Mongoexport y Mongoimport.....                  | 39 |
| 4.1.Mongoexport.....                                     | 40 |
| 4.2.Mongoimport.....                                     | 41 |

## 1.Estructuras de colecciones.

Mi base de datos consta de una colección principal llamada ventas de la cual salen las colecciones con las que trabajamos.

```
{
  idVenta: 3,
  items: [
    {
      idOrdenador: 31,
      características: {
        placa: {
          socket: "am4",
          marca: "aorus",
          puertos: [
            "vga",
            "hdmi",
            "thundervolt",
            "usb3.0",
            "rj45",
            "jack"
          ],
          formato: "micro atx"
        },
        memoria: {
          capacidad: 16,
          tipo: "ddr4"
        },
        procesador: {
          marca: "amd",
          modelo: "ryzen 7 3800x",
          generacion: 3
        },
        tarjeta_grafica: {
          marca: "amd",
          modelo: "radeon 5700xt"
        },
        caja: {
          color: ["negro", "verde"],
          dimensiones: {h: 70, w: 55 },
          usb: [ 3.1, 4.0 ]
        },
        fuente: "750w",
        disco: [
          {
            tipo: "ssd",
            capacidad: 512
          },
          {
            tipo: "hdd",
            capacidad: 512
          }
        ],
        sistema_operativo: false,
        fecha_fabricacion: new Date("2021-02-22")
      },
      perifericos: [
        "Teclado Krom Kernel",
        "Raton Logitech G502",
        "Pantalla AOC"
      ],
      precio: 1300.10
    }
  ],
  vendedor: "Mediamark",
  comprador: "Particular: Joaquin Jimenez",
  fechaVenta: new Date("2022-1-2")
},
```

Ese es un ejemplo de documento de la colección “ventas”, consta de un campo con el id de la venta, un campo items tipo array de documentos donde vemos el id del ordenador, las características, periféricos y el precio, (en este caso solo hay un documento pero puede llegar a haber más de uno) un campo vendedor y comprador, y la fecha de venta.

De esta colección sacamos 4 colecciones distintas, una colección soldItems donde vemos los ordenadores que se han vendido, una colección sales donde vemos la información de las ventas, una colección providers donde guardamos la información de los proveedores y una colección products donde guardamos la información de los productos que usamos para armar los ordenadores.

## 1.1.Colección soldItems.

Conseguimos esta colección mediante un \$out que expresamos en el siguiente aggregate:

```
db.ventas.aggregate([
  {
    $unwind: {
      path: "$items"
    }
  },
  {
    $project: {
      _id: 0,
      _idOrdenador: "$items.idOrdenador",
      idVenta:1,
      caracteristicas: "$items.caracteristicas",
      perifericos:"$items.perifericos",
      precio:"$items.precio"
    }
  },
  {
    $out: "soldItems"
  }
])
```

Realizamos un \$unwind del campo items para tener un documento por cada ordenador y proyectamos solo el id, las características, el id de la venta, los periféricos y el precio.

El resultado de un documento de este \$out es el siguiente:

```
"_id":{
  "$oid":"6224b83ab0c9fb3d4bc6ac0f"
},
"idVenta":1.0,
"_idOrdenador":11.0,
"caracteristicas":{
  "placa":{
    "socket":"am4",
    "marca":"gigabyte",
    "puertos":[
      "vga",
      "hdmi",
      "usb2.0",
      "rj45",
      "jack"
    ],
    "formato":"atx"
  },
  "memoria":{
    "capacidad":16.0,
    "tipo":"ddr4"
  },
  "procesador":{
    "marca":"amd",
    "modelo":"ryzen 5 2500 ",
    "generacion":2.0
  },
  "tarjeta_grafica":{
    "tipo":"integrada"
  },
  "caja":{
    "color":[
      "balnco",
      "azul"
    ],
    "dimensiones":{
      "h":80.0,
      "w":65.0
    },
    "usb":[3.0,3.1]
  },
  "fuente":"650w",
  "disco":[
    {
      "tipo":"hdd",
      "capacidad":1024.0
    },
    {
      "tipo":"ssd",
      "capacidad":256.0
    }
  ],
  "sistema_operativo":true,
  "fecha_fabricacion":{
    "$date":"2018-05-12T00:00:00Z"
  }
},
"perifericos":[
  "Teclado Krom Kernel",
  "Raton Logitech G203",
  "Pantalla AOC"
],
"precio":900.6
```

## 1.2.Colección sales.

Conseguimos esta colección mediante un \$out que expresamos en el siguiente aggregate:

```
db.ventas.aggregate( [  
  {  
    $project: {  
      _id: 0,  
      items: 0,  
    }  
  },  
  {  
    $out: "sales"  
  }  
])
```

Realizamos un \$project en el que quitemos el \_id y el campo items de manera que se quede el solo la información de las ventas.

El resultado de un documento de este \$out es el siguiente:

```
{  
  "_id":{  
    "$oid":"6224b841b0c9fb3d4bc6ac26"  
  },  
  "idVenta":1.0,  
  "vendedor":"Real Microinformatica",  
  "comprador":"Instituto Jesus del Granpoder",  
  "fechaVenta":{  
    "$date":"2022-02-22T00:00:00Z"  
  }  
}
```

## 1.3.Colección providers.

Conseguimos esta colección mediante un \$out que expresamos en el siguiente aggregate:

```

db.ventas.aggregate( [
  {
    $project: {
      name: {
        $concatArrays:[
          "$items.caracteristicas.placa.marca",
          "$items.caracteristicas.procesador.marca",
          "$items.caracteristicas.tarjeta_grafica.marca"
        ]
      }
    }
  },
  {
    $unwind: {
      path: "$name"
    }
  },
  {
    $group: {
      _id: "$name",
      noProvisiones: {
        $sum: 1
      }
    }
  },
  {
    $project:{
      _id: 0,
      name: "$_id",
      noProvisiones: "$noProvisiones"
    }
  },
  {
    $out: "providers"
  }
])

```

Hacemos un \$project creando un nuevo campo que se llame name donde guardaremos un array con el nombre de todas las marcas usadas para montar los equipos de manera que guardemos todas las marcas que usa cada equipo dentro de un array.

Después hacemos un \$unwind para tener un documento con cada marca, un \$group para agrupar por marca y creamos un campo número de provisiones para contar el número de veces que nos ha dado componentes ese proveedor, un \$project para que el nombre que al hacer el group estaba en el campo \_id sea otra vez el campo nombre y el campo numero de provisiones se quede igual.

Un ejemplo de un documento de esta colección es el siguiente:

```
{
  "_id":{
    "$oid":"6224b849b0c9fb3d4bc6ac35"
  },
  "name":"msi",
  "noProvisiones":3.0
}
```

## 1.4.Colección products.

Para esta colección he usado 4 aggregate distintos ya que los productos están en distintos campos del documento. Primero he creado un aggregate para los procesadores:

```
db.ventas.aggregate( [
  {
    $unwind: {
      path: "$items"
    }
  },
  {
    $lookup: {
      from: 'providers',
      localField: 'items.caracteristicas.procesador.marca',
      foreignField: 'name',
      as: 'idmarca'
    }
  },
  {
    $project: {
      name: {
        $concat:[
          "$items.caracteristicas.procesador.marca",
          " ",
          "$items.caracteristicas.procesador.modelo"
        ]
      },
      idMarca: {
        $arrayElemAt: [
          "$idmarca._id",
          0
        ]
      }
    }
  },
  {
    $group: {
      _id: {
        name:"$name",
        idMarca:"$idMarca"
      },
      noArticulos: {
        $sum: 1
      }
    }
  },
  {
    $out: "products"
  }
])
```



En este aggregate realizamos un \$unwind de items para conseguir un documento de cada ordenador, luego un \$lookup a providers para conseguir el id de la marca que es un campo que queremos que esté en la colección. Luego hacemos un \$project donde creamos el campo name que sera el nombre del procesador (ej: Intel i7 900k) usando \$concat y concatenando la marca y el modelo, en el id de la marca hacemos un \$arrayElemAt para sacar el id del array. Seguidamente agrupamos por nombre del producto y id de la marca y calculamos el número de artículos que hay de ese producto.

Realizamos el mismo procedimiento con la placa base, la tarjeta gráfica y los periféricos.

```
db.ventas.aggregate( [
  {
    $unwind: {
      path: "$items"
    }
  },
  {
    $lookup: {
      from: 'providers',
      localField: 'items.caracteristicas.tarjeta_grafica.marca',
      foreignField: 'name',
      as: 'idmarca'
    }
  },
  {
    $project: {
      name: {
        $concat:[
          "$items.caracteristicas.tarjeta_grafica.marca",
          " ",
          "$items.caracteristicas.tarjeta_grafica.modelo"
        ]
      },
      idMarca: {
        $arrayElemAt: [
          "$idmarca._id",
          0
        ]
      }
    }
  },
  {
    $group: {
      _id: {
        name:"$name",
        idMarca:"$idMarca"
      },
      noArticulos: {
        $sum: 1
      }
    }
  },
  {
    $merge: {
      into: "products",
      whenMatched: "merge"
    }
  }
])
```

En esta consulta en lugar de \$out usamos \$merge para que en lugar de sobrescribir la colección products, añada los datos a la colección sin eliminar lo demás.

```
db.ventas.aggregate( [
  {
    $unwind: {
      path: "$items"
    }
  },
  {
    $lookup: {
      from: 'providers',
      localField: 'items.caracteristicas.placa.marca',
      foreignField: 'name',
      as: 'idmarca'
    }
  },
  {
    $project: {
      name: {
        $concat:[
          "$items.caracteristicas.placa.marca",
          " ",
          "socket",
          " ",
          "$items.caracteristicas.placa.socket",
          " ",
          "$items.caracteristicas.placa.formato"
        ]
      },
      idMarca: {
        $arrayElemAt: [
          "$idmarca._id",
          0
        ]
      }
    }
  },
  {
    $group: {
      _id: {
        name: "$name",
        idMarca: "$idMarca"
      },
      noArticulos: {
        $sum: 1
      }
    }
  },
  {
    $merge: {
      into: "products",
      whenMatched: "merge"
    }
  }
])
```

```
db.ventas.aggregate( [
  {
    $unwind: {
      path: "$items"
    }
  },
  {
    $project: {
      name: "$items.perifericos",
      idMarca: null
    }
  },
  {
    $unwind: {
      path: "$name"
    }
  },
  {
    $group: {
      _id: {
        name: "$name",
        idMarca: "$idMarca"
      },
      noArticulos: {
        $sum: 1
      }
    }
  },
  {
    $merge: {
      into: "products",
      whenMatched: "merge"
    }
  }
])
```

En este último usamos los periféricos, con id de marca en null ya que no tenemos registro de las marcas de los periféricos

Un ejemplo de documento en esta colección es el siguiente:

```
{
  "_id":{
    "name":"intel i5 6500k",
    "idMarca":{
      "$oid":"6224b849b0c9fb3d4bc6ac3d"
    }
  },
  "noArticulos":1.0
}
```

```
{ "_id":{"name":"amd ryzen 5 2500","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":1.0}
{ "_id":{"name":"amd ryzen 9 5950x","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":1.0}
{ "_id":{"name":"intel i5 8500k","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3d"}}, "noArticulos":1.0}
{ "_id":{"name":"intel i9 9900k","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3d"}}, "noArticulos":1.0}
{ "_id":{"name":"amd athlon","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":2.0}
{ "_id":{"name":"intel pentium","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3d"}}, "noArticulos":1.0}
{ "_id":{"name":"intel i3 6300k","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3d"}}, "noArticulos":2.0}
{ "_id":{"name":"amd ryzen 7 4550x","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":1.0}
{ "_id":{"name":"intel i7 11700k","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3d"}}, "noArticulos":1.0}
{ "_id":{"name":"intel i9 12900k","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3d"}}, "noArticulos":1.0}
{ "_id":{"name":"amd ryzen 5 2500 ","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":1.0}
{ "_id":{"name":"intel i7 9700k","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3d"}}, "noArticulos":2.0}
{ "_id":{"name":"amd ryzen 7 3700x","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":1.0}
{ "_id":{"name":"amd ryzen 3 1300","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":1.0}
{ "_id":{"name":"amd ryzen 7 2700x","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":1.0}
{ "_id":{"name":"amd ryzen 7 3800x","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":1.0}
{ "_id":{"name":"intel i9 12900kf","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3d"}}, "noArticulos":1.0}
{ "_id":{"name":"nvidia gtx 1060ti","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3a"}}, "noArticulos":1.0}
{ "_id":{"name":"nvidia rtx 3090","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3a"}}, "noArticulos":2.0}
{ "_id":{"name":"amd radeon 5700xt","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":1.0}
{ "_id":{"name":"nvidia gtz 1050ti","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3a"}}, "noArticulos":1.0}
{ "_id":{"name":"nvidia rtx 2080ti","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3a"}}, "noArticulos":1.0}
{ "_id":{"name":"nvidia rtx 2097ti","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3a"}}, "noArticulos":1.0}
{ "_id":{"name":"nvidia rtx 3070 super","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3a"}}, "noArticulos":2.0}
{ "_id":{"name":"amd radeon 5500xt","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":1.0}
{ "_id":{"name":"amd radeon rx590","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3c"}}, "noArticulos":2.0}
{ "_id":{"name":"nvidia rtx 3080ti","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3a"}}, "noArticulos":1.0}
{ "_id":{"name":"nvidia rtx 3090ti","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3a"}}, "noArticulos":1.0}
{ "_id":{"name":"asus socket lga 1700 atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac36"}}, "noArticulos":1.0}
{ "_id":{"name":"asrock socket lga 1150 micro atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac39"}}, "noArticulos":1.0}
{ "_id":{"name":"asus socket lga 1151 atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac36"}}, "noArticulos":1.0}
{ "_id":{"name":"aorus socket am4 atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac38"}}, "noArticulos":1.0}
{ "_id":{"name":"msi socket lga 1700 atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac35"}}, "noArticulos":1.0}
{ "_id":{"name":"asus socket am3 micro atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac36"}}, "noArticulos":1.0}
{ "_id":{"name":"gigabyte socket am4 atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac37"}}, "noArticulos":2.0}
{ "_id":{"name":"msi socket am4 micro atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac35"}}, "noArticulos":2.0}
{ "_id":{"name":"asrock socket lga 1150 atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac39"}}, "noArticulos":2.0}
{ "_id":{"name":"hzxt socket am4 micro atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac3b"}}, "noArticulos":1.0}
{ "_id":{"name":"gigabyte socket am3 micro atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac37"}}, "noArticulos":1.0}
{ "_id":{"name":"gigabyte socket lga 1151 micro atx","idMarca":{"$oid":"6224b849b0c9fb3d4bc6ac37"}}, "noArticulos":1.0}
```

## 2. Uso de Mongo Atlas.

Debemos crearnos una cuenta en Mongo Atlas, en mi caso me he registrado con Google. Una vez tengamos la cuenta nos aparecerá este menú:



Ahora creamos una nueva base de datos de Atlas, para ello debemos crear un proyecto y pulsar en el botón crear para crear una base de datos.



Creamos un cluster compartido con amazon web services.

## Crear un clúster compartido

**¡Bienvenido a MongoDB Atlas !** Hemos recomendado algunas de nuestras opciones más populares, pero no dude en personalizar su clúster según sus necesidades. Para más información, consulta nuestra [documentación](#).

AVANCE sin servidor

Dedicado

GRATIS Compartido

Para aprender y explorar MongoDB en un entorno sandbox. Controles básicos de configuración.

No se requiere tarjeta de crédito para comenzar. Actualice a clústeres dedicados para una funcionalidad completa. Explore con conjuntos de datos de muestra. Límite de un clúster gratuito por proyecto.

**Proveedor de nube y región** AWS, París (eu-west-3) ▾

aws

Google Cloud

Azure

★Región recomendada ⓘ ⓘRegión de nivel pagado ⓘ

| NORTEAMÉRICA  | EUROPA   | AUSTRALIA   |
|---|--|---|
| <div>🇺🇸 Virginia del Norte ( us-east-1 ) ★</div> <div>🇺🇸 Oregón ( us-west-2 ) ★</div> | <div>🇫🇷 París ( eu-west-3 ) ★</div> <div>🇩🇪 Fráncfort ( eu-central-1 ) ★</div> | <div>🇦🇺 Sidney ( ap-sureste-2 ) ★</div> <div>ASIA</div> |

**GRATIS** ¡Siempre libre! Su clúster M0 es ideal para experimentar en un espacio aislado limitado. Puede actualizar a un clúster de producción en cualquier momento. [Atrás](#) [Crear clúster](#)

Crearemos un usuario y una contraseñas para poder acceder a la base de datos.

### 1 ¿Cómo le gustaría autenticar su conexión?

Su primer usuario tendrá permiso para leer y escribir cualquier dato en su proyecto.

Nombre de usuario y contraseña

Certificado

Cree un usuario de la base de datos utilizando un nombre de usuario y una contraseña. Los usuarios tendrán [privilegios de lectura y escritura en cualquier base de datos](#) de forma predeterminada. Puede actualizar estos permisos y/o crear usuarios adicionales más adelante. Asegúrese de que estas credenciales sean diferentes a su nombre de usuario y contraseña de MongoDB Cloud.

**Nombre de usuario**

gonzalo

**Clave** 🔑

.....

[Autogenerar contraseña segura](#) [Dupdo](#)

[Crear usuario](#)

Añadimos también nuestra dirección IP actual para poder acceder a la base de datos.

Agregue entradas a su lista de acceso IP

Solo una dirección IP que agregue a su lista de acceso podrá conectarse a los clústeres de su proyecto. Puede administrar las entradas de IP existentes a través de la [página de acceso a la red](#).

| Dirección IP   | Descripción   |                                 |   |
|--|---|---------------------------------|---|
| <input type="text" value="Ingrese la dirección IP"/> | <input type="text" value="Introduce la descripción"/> | <button>Añadir entrada</button> | <button>Agregar mi dirección IP actual</button> |

---

| Lista de acceso IP | Descripción     |                           |
|--------------------|-----------------|---------------------------|
| <div></div>        | Mi dirección IP | <button>ELIMINAR</button> |

Ahora en el apartado de base de datos veremos nuestra base de datos ya creada.

GONZALO'S ORG - 2023-02-15 > PROYECTO2TRIMESTRE

### Database Deployments

+ Create

Cluster0
Connect
View Monitoring
Browse Collections
...
FREE
SHARED

● R 0
● W 0
Last 46 seconds

● In 0.0 B/s
● Out 0.0 B/s
Last 46 seconds

● Data Size 0.0 B
Last 46 seconds


● Enhance Your Experience


For production throughput and richer metrics, upgrade to a dedicated cluster now!


Upgrade

| VERSION | REGION                  | CLUSTER TIER         | TYPE                  | BACKUPS  | LINKED REALM APP | ATLAS SEARCH                 |
|---------|-------------------------|----------------------|-----------------------|----------|------------------|------------------------------|
| 5.0.6   | AWS / Paris (eu-west-3) | M0 Sandbox (General) | Replica Set - 3 nodes | Inactive | None Linked      | <a href="#">Create Index</a> |

Para conectarnos solo tenemos que pulsar en connect y nos aparecerán distintas opciones para conectarte.

**Connect with the MongoDB Shell**  
Interact with your cluster using MongoDB's interactive Javascript interface >

**Connect your application**  
Connect your application to your cluster using MongoDB's native drivers >

**Connect using MongoDB Compass**  
Explore, modify, and visualize your data with MongoDB's GUI >

Podemos conectarnos con mongo shell o con mongo compass.

## 2.1. Conectar una base de datos usando mongo shell.

Para conectarte con mongo shell debes usar un comando con la siguiente sintaxis que aparece en la página:

```
mongosh "mongodb+srv://cluster0.segc7.mongodb.net/myFirstDatabase" --apiVersion 1  
--username gonzalo
```

Ejecutamos este comando en la PowerShell y accedemos a la base de datos.

```
PS C:\Users\grans> mongosh "mongodb+srv://cluster0.segc7.mongodb.net/myFirstDatabase" --apiVersion 1 --username gonzalo  
Enter password: *****  
Current Mongosh Log ID: 6224fe5699c5a81614876a9a  
Connecting to:      mongodb+srv://cluster0.segc7.mongodb.net/myFirstDatabase  
Using MongoDB:      5.0.6 (API Version 1)  
Using Mongosh:      1.1.0  
  
For mongosh info see: https://docs.mongodb.com/mongodb-shell/  
  
To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).  
You can opt-out by running the disableTelemetry() command.  
  
Warning: Found ~/.mongorc.js, but not ~/.mongoshrc.js. ~/.mongorc.js will not be loaded.  
You may want to copy or rename ~/.mongorc.js to ~/.mongoshrc.js.  
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> _
```

Como vemos hemos entrado a la base de datos sin problemas.

## 2.2. Conectar una base de datos usando mongo compass.

Debemos poner en nueva conexión en enlace qu aparece en Mongo Atlas :

```
mongodb+srv://gonzalo:<password>@cluster0.segc7.mongodb.net/test
```

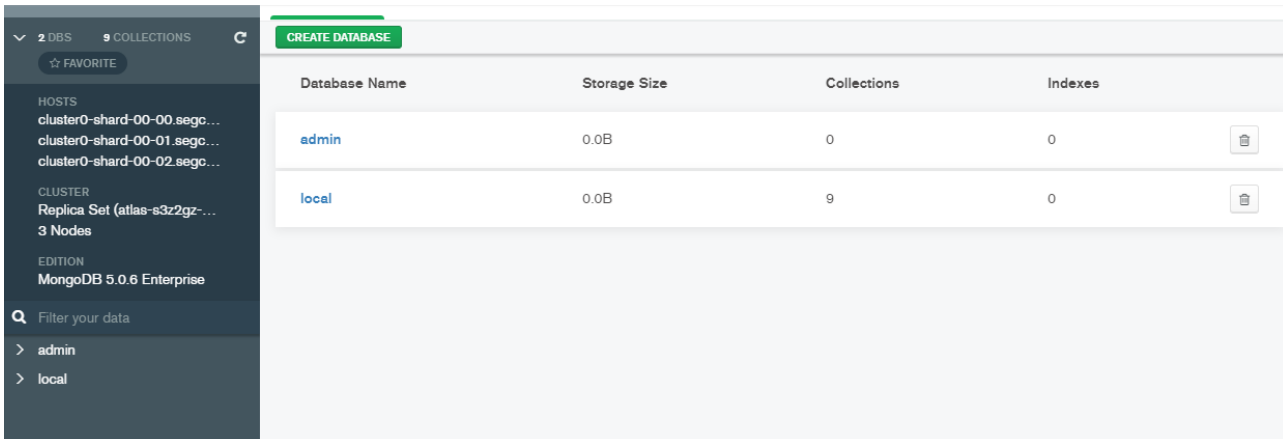


Paste your connection string (SRV or Standard ⓘ)

mongodb+srv://gonzalo:[REDACTED]@cluster0.segc7.mongodb.net/test

Connect

Como vemos accedemos a la base de datos sin ningún problema.



| Database Name | Storage Size | Collections | Indexes |
|---------------|--------------|-------------|---------|
| admin         | 0.0B         | 0           | 0       |
| local         | 0.0B         | 9           | 0       |

Yo trabajaré con Mongo Atlas desde Mongo Shell.

## 3.Creación de la base de datos.

### 3.1.Inserts.

Realizamos los inserts en la nueva base de datos.

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("6225317fad8f9ab2d9cc97a8"),
    '1': ObjectId("6225317fad8f9ab2d9cc97a9"),
    '2': ObjectId("6225317fad8f9ab2d9cc97aa"),
    '3': ObjectId("6225317fad8f9ab2d9cc97ab"),
    '4': ObjectId("6225317fad8f9ab2d9cc97ac"),
    '5': ObjectId("6225317fad8f9ab2d9cc97ad"),
    '6': ObjectId("6225317fad8f9ab2d9cc97ae"),
    '7': ObjectId("6225317fad8f9ab2d9cc97af"),
    '8': ObjectId("6225317fad8f9ab2d9cc97b0")
  }
}
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> _
```

Hemos realizado los inserts pero nos dicen que en el ultimo pedido falta un ordenador por registrarse a la base de datos, lo metemos usando updateOne y \$push.

```
db.ventas.updateOne(
  {
    idVenta: 9
  },
  {
    $push: {
      items: {
        idOrdenador: 93,
        características: {
          placa: {
            socket: "lga 1151",
            marca: "hzxt",
            puertos: [
              "hdmi",
              "dvi",
              "rj45",
              "usb4.0",
              "jack"
            ],
            formato: "micro atx"
          },
          memoria: {
            capacidad: 32,
            tipo: "ddr4"
          },
          procesador: {
            marca: "intel",
            modelo: "i7 9700k",
            generacion: 9
          },
          tarjeta_grafica: {
            marca: "nvidia",
            modelo: "rtx 3090ti"
          },
          caja: {
            color: ["blanco"],
            dimensiones: {h:80, w:75 },
            usb: [ 3.2, 3.1 ]
          },
          fuente: "850w",
          disco: [
            {
              tipo: "ssd",
              capacidad: 512
            },
            {
              tipo: "hdd",
              capacidad: 512
            }
          ],
          sistema_operativo: false,
          fecha_fabricacion: new Date("2019-12-02")
        },
        perifericos: ["Teclado Corsair", "Raton Logitech G502"],
        precio: 2700.00
      }
    }
  }
)
```

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> █
```

Ya tendríamos los datos en la base de datos.

### 3.2.Outs.

Ahora ejecutamos los \$outs en la base de datos para crear las distintas colecciones. Creamos soldItems:

```
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> db.ventas.aggregate( [
...   {
...     $unwind: {
...       path: "$items"
...     }
...   },
...   {
...     $project: {
...       _id: 0,
...       _idOrdenador: "$items.idOrdenador",
...       idVenta:1,
...       características: "$items.características",
...       perifericos:"$items.perifericos",
...       precio:"$items.precio"
...     }
...   },
...   {
...     $out: "soldItems"
...   }
... ] )
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase>
```

Creamos sales:

```
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> db.ventas.aggregate( [
...   {
...     $project: {
...       _id: 0,
...       items: 0,
...     }
...   },
...   {
...     $out: "sales"
...   }
... ] )
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase>
```

Creamos providers:

```
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> db.ventas.aggregate( [
...   {
...     $project: {
...       name: {
...         $concatArrays:[
...           "$items.caracteristicas.placa.marca",
...           "$items.caracteristicas.procesador.marca",
...           "$items.caracteristicas.tarjeta_grafica.marca"
...         ]
...       }
...     },
...     $unwind: {
...       path: "$name"
...     },
...     $group: {
...       _id: "$name",
...       noProvisiones: {
...         $sum: 1
...       }
...     },
...     $project:{
...       _id: 0,
...       name: "$_id",
...       noProvisiones: "$noProvisiones"
...     },
...     $out: "providers"
...   ] )

Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> _
```

Creamos products:

```
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> db.ventas.aggregate( [
...   {
...     $unwind: {
...       path: "$items"
...     }
...   },
...   {
...     $lookup: {
...       from: 'providers',
...       localField: 'items.características.procesador.marca',
...       foreignField: 'name',
...       as: 'idmarca'
...     }
...   },
...   {
...     $project: {
...       name: {
...         $concat:[
...           "$items.características.procesador.marca",
...           " ",
...           "$items.características.procesador.modelo"
...         ]
...       },
...       idMarca: {
...         $arrayElemAt: [
...           "$idmarca._id",
...           0
...         ]
...       }
...     }
...   },
...   {
...     $group: {
...       _id: {
...         name:"$name",
...         idMarca:"$idMarca"
...       },
...       noArticulos: {
...         $sum: 1
...       }
...     }
...   },
...   {
...     $out: "products"
...   }
... ])
```

Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> █

```

Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> db.ventas.aggregate( [
...   {
...     $unwind: {
...       path: "$items"
...     }
...   },
...   {
...     $lookup: {
...       from: 'providers',
...       localField: 'items.caracteristicas.procesador.marca',
...       foreignField: 'name',
...       as: 'idmarca'
...     }
...   },
...   {
...     $project: {
...       name: {
...         $concat:[
...           "$items.caracteristicas.procesador.marca",
...           " ",
...           "$items.caracteristicas.procesador.modelo"
...         ]
...       },
...       idMarca: {
...         $arrayElemAt: [
...           "$idmarca._id",
...           0
...         ]
...       }
...     }
...   },
...   {
...     $group: {
...       _id: {
...         name: "$name",
...         idMarca: "$idMarca"
...       },
...       noArticulos: {
...         $sum: 1
...       }
...     }
...   },
...   {
...     $out: "products"
...   }
... ])

...   {
...     $unwind: {
...       path: "$items"
...     }
...   },
...   {
...     $lookup: {
...       from: 'providers',
...       localField: 'items.caracteristicas.tarjeta_grafica.marca',
...       foreignField: 'name',
...       as: 'idmarca'
...     }
...   },
... ]

```

```

... {
..... $project: {
.....     name: {
.....         $concat:[
.....             "$items.caracteristicas.tarjeta_grafica.marca",
.....             " ",
.....             "$items.caracteristicas.tarjeta_grafica.modelo"
.....         ]
.....     },
.....     idMarca: {
.....         $arrayElemAt: [
.....             "$idmarca._id",
.....             0
.....         ]
.....     }
..... },
..... {
.....     $group: {
.....         _id: {
.....             name:"$name",
.....             idMarca:"$idMarca"
.....         },
.....         noArticulos: {
.....             $sum: 1
.....         }
.....     }
..... },
..... {
.....     $merge: {
.....         into: "products",
.....         whenMatched: "merge"
.....     }
..... }
... ]}

Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> _

```

```

Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> db.ventas.aggregate( [
...   {
.....     $unwind: {
.....       path: "$items"
.....     }
.....   },
...   {
.....     $lookup: {
.....       from: 'providers',
.....       localField: 'items.caracteristicas.placa.marca',
.....       foreignField: 'name',
.....       as: 'idmarca'
.....     }
.....   },
...   {
.....     $project: {
.....       name: {
.....         $concat:[
.....           "$items.caracteristicas.placa.marca",
.....           " ",
.....           "socket",
.....           " ",
.....           "$items.caracteristicas.placa.socket",
.....           " ",
.....           "$items.caracteristicas.placa.formato"
.....         ]
.....       },
.....       idMarca: {
.....         $arrayElemAt: [
.....           "$idmarca._id",
.....           0
.....         ]
.....       }
.....     }
.....   },
...   {
.....     $group: {
.....       _id: {
.....         name:"$name",
.....         idMarca:"$idMarca"
.....       },
.....       noArticulos: {
.....         $sum: 1
.....       }
.....     }
.....   },
...   {
.....     $merge: {
.....       into: "products",
.....       whenMatched: "merge"
.....     }
.....   }
... ] )

Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase>

```



```
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> db.ventas.aggregate( [
...   {
.....     $unwind: {
.....       path: "$items"
.....     }
.....   },
...   {
.....     $project: {
.....       name: "$items.perifericos",
.....       idMarca: null
.....     }
.....   },
...   {
.....     $unwind: {
.....       path: "$name"
.....     }
.....   },
...   {
.....     $group: {
.....       _id: {
.....         name: "$name",
.....         idMarca: "$idMarca"
.....       },
.....       noArticulos: {
.....         $sum: 1
.....       }
.....     }
.....   },
...   {
.....     $merge: {
.....       into: "products",
.....       whenMatched: "merge"
.....     }
.....   }
... ])
```

```
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> _
```

Comprobamos que las colecciones se han añadido correctamente.

```
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> show collections
products
providers
sales
soldItems
ventas
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> _
```

### 3.3.Queries.

Queremos saber cual es el ordenador más caro y el mas barato.

```
db.soldItems.aggregate( [
  {
    $group: {
      _id: null,
      pcMasCaro: {
        $max: "$precio"
      },
      pcMasBarato: {
        $min: "$precio"
      }
    }
  }
] )
```

```
[ { _id: null, pcMasCaro: 5500, pcMasBarato: 500 } ]
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> _
```

Queremos ver todos los productos de cada marca.

```
db.providers.aggregate([
  {
    $lookup: {
      from: 'products',
      localField: '_id',
      foreignField: '_id.idMarca',
      as: 'productos'
    }
  },
  {
    $project: {
      name: 1,
      productos: "$productos._id.name"
    }
  }
])
```

```
[
  {
    _id: ObjectId("62253774bae29ac5a703252a"),
    name: 'asrock',
    productos: [
      'asrock socket lga 1150 atx',
      'asrock socket lga 1150 micro atx'
    ]
  },
  {
    _id: ObjectId("62253774bae29ac5a703252b"),
    name: 'intel',
    productos: [
      'intel i5 8500k',
      'intel i7 11700k',
      'intel i9 12900kf',
      'intel i9 9900k',
      'intel pentium',
      'intel i3 6500k',
      'intel i7 9700k',
      'intel i9 12900k',
      'intel i5 6500k'
    ]
  },
  {
    _id: ObjectId("62253774bae29ac5a703252c"),
    name: 'nvidia',
    productos: [
      'nvidia rtx 3070 super',
      'nvidia gtx 1080ti',
      'nvidia rtx 2080ti',
      'nvidia rtx 3090',
      'nvidia rtx 2087ti',
      'nvidia gtx 1080ti',
      'nvidia rtx 3080ti',
      'nvidia rtx 3090ti'
    ]
  },
  {
    _id: ObjectId("62253774bae29ac5a703252d"),
    name: 'msi',
    productos: [ 'msi socket lga 1700 atx', 'msi socket am4 micro atx' ]
  },
  {
    _id: ObjectId("62253774bae29ac5a703252e"),
    name: 'asus',
    productos: [
      'asus socket lga 1151 atx',
      'asus socket lga 1700 atx',
      'asus socket am3 micro atx'
    ]
  },
  {
    _id: ObjectId("62253774bae29ac5a703252f"),
    name: 'aorus',
    productos: [
      'aorus socket lga 1151 atx',
      'aorus socket lga 1700 micro atx',
      'aorus socket am4 micro atx',
      'aorus socket am4 atx'
    ]
  },
  {
    _id: ObjectId("62253774bae29ac5a7032530"),
    name: 'amd',
    productos: [
      'amd ryzen 3 1300',
      'amd ryzen 7 3800x',
      'amd ryzen 5 2500',
      'amd ryzen 9 5950x',
      'amd ryzen 7 2700x',
      'amd athlon',
      'amd ryzen 7 4850x',
      'amd ryzen 7 3700x',
      'amd ryzen 5 2500',
      'amd radeon 5700xt',
      'amd radeon 5800xt',
      'amd radeon rx590'
    ]
  },
  {
    _id: ObjectId("62253774bae29ac5a7032531"),
    name: 'gigabyte',
    productos: [
      'gigabyte socket am4 atx',
      'gigabyte socket lga 1151 micro atx',
      'gigabyte socket am3 micro atx'
    ]
  },
  {
    _id: ObjectId("62253774bae29ac5a7032532"),
    name: 'hzxt',
    productos: [ 'hzxt socket am4 micro atx', 'hzxt socket lga 1151 micro atx' ]
  }
]
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase>
```

Queremos mostrar la media del precio de un ordenador agrupada por meses de un año.

```
db.soldItems.aggregate([
  {
    $group : {
      _id : {
        $dateToString: {
          format: "%Y-%m",
          date: "$caracteristicas.fecha_fabricacion"
        }
      },
      mediaPrecios: {
        $avg: "$precio"
      }
    }
  },
  {
    $sort: {
      _id: -1
    }
  }
])
```

```
[
  { _id: '2017-06', mediaPrecios: 700.5 },
  { _id: '2020-07', mediaPrecios: 2000.99 },
  { _id: '2018-09', mediaPrecios: 1200.1 },
  { _id: '2019-11', mediaPrecios: 750.4 },
  { _id: '2010-02', mediaPrecios: 500 },
  { _id: '2015-10', mediaPrecios: 500.5 },
  { _id: '2019-05', mediaPrecios: 1400.5 },
  { _id: '2019-12', mediaPrecios: 2550.025 },
  { _id: '2021-02', mediaPrecios: 1300.1 },
  { _id: '2014-11', mediaPrecios: 800.89 },
  { _id: '2020-09', mediaPrecios: 3000.99 },
  { _id: '2017-09', mediaPrecios: 800.1 },
  { _id: '2015-05', mediaPrecios: 800 },
  { _id: '2021-10', mediaPrecios: 5500 },
  { _id: '2020-12', mediaPrecios: 1200.1 },
  { _id: '2018-05', mediaPrecios: 900.6 },
  { _id: '2019-04', mediaPrecios: 1200.34 },
  { _id: '2016-01', mediaPrecios: 800.99 },
  { _id: '2021-11', mediaPrecios: 2800.99 }
]
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase>
```

Queremos mostrar el ordenador más caro y el más barato fabricado en 2021.

```

db.soldItems.aggregate([
  {
    $match : {
      "caracteristicas.fecha_fabricacion": {
        $gte: new ISODate("2021-01-01"),
        $lt: new ISODate("2022-01-01")
      }
    },
  },
  {
    $group : {
      _id : null,
      ordenadorMasCaro: {
        $max: "$precio"
      },
      ordenadorMasBarato: {
        $min: "$precio"
      },
    },
  },
])

```

```

[ { _id: null, ordenadorMasCaro: 5500, ordenadorMasBarato: 1300.1 } ]
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase>

```

Queremos saber cuantas unidades de cada articulo se ha vendido de cada vendedor.

```

db.providers.aggregate( [
  {
    $lookup: {
      from: 'products',
      localField: '_id',
      foreignField: '_id.idMarca',
      as: 'productos'
    }
  },
  {
    $unwind: {
      path: "$productos"
    }
  },
  {
    $project: {
      _id: 0,
      proveedor: "$name",
      producto: "$productos._id.name",
      noVentas: "$productos.noArticulos"
    }
  },
] )

```

```

{
  proveedor: 'asrock',
  producto: 'asrock socket lga 1150 atx',
  noVentas: 2
},
{
  proveedor: 'asrock',
  producto: 'asrock socket lga 1150 micro atx',
  noVentas: 1
},
{ proveedor: 'intel', producto: 'intel i5 8500k', noVentas: 1 },
{ proveedor: 'intel', producto: 'intel i7 11700k', noVentas: 1 },
{ proveedor: 'intel', producto: 'intel i9 12900kf', noVentas: 1 },
{ proveedor: 'intel', producto: 'intel i9 9900k', noVentas: 1 },
{ proveedor: 'intel', producto: 'intel pentium', noVentas: 1 },
{ proveedor: 'intel', producto: 'intel i3 6300k', noVentas: 2 },
{ proveedor: 'intel', producto: 'intel i7 9700k', noVentas: 2 },
{ proveedor: 'intel', producto: 'intel i9 12900k', noVentas: 1 },
{ proveedor: 'intel', producto: 'intel i5 6500k', noVentas: 1 },
{
  proveedor: 'nvidia',
  producto: 'nvidia rtx 3070 super',
  noVentas: 2
},
{ proveedor: 'nvidia', producto: 'nvidia gtx 1060ti', noVentas: 1 },
{ proveedor: 'nvidia', producto: 'nvidia rtx 2080ti', noVentas: 1 },
{ proveedor: 'nvidia', producto: 'nvidia rtx 3090', noVentas: 2 },
{ proveedor: 'nvidia', producto: 'nvidia rtx 2097ti', noVentas: 1 },
{ proveedor: 'nvidia', producto: 'nvidia gtx 1050ti', noVentas: 1 },
{ proveedor: 'nvidia', producto: 'nvidia rtx 3080ti', noVentas: 1 },
{ proveedor: 'nvidia', producto: 'nvidia rtx 3090ti', noVentas: 1 },
{
  proveedor: 'msi',
  producto: 'msi socket lga 1700 atx',
  noVentas: 1
}

```

```

Type "it" for more
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> it
[
  {
    proveedor: 'msi',
    producto: 'msi socket am4 micro atx',
    noVentas: 2
  },
  {
    proveedor: 'asus',
    producto: 'asus socket lga 1151 atx',
    noVentas: 1
  },
  {
    proveedor: 'asus',
    producto: 'asus socket lga 1700 atx',
    noVentas: 1
  },
  {
    proveedor: 'asus',
    producto: 'asus socket am3 micro atx',
    noVentas: 1
  },
  {
    proveedor: 'aorus',
    producto: 'aorus socket lga 1151 atx',
    noVentas: 1
  },
  {
    proveedor: 'aorus',
    producto: 'aorus socket lga 1700 micro atx',
    noVentas: 1
  },
  {
    proveedor: 'aorus',
    producto: 'aorus socket am4 micro atx',
    noVentas: 2
  },
  { proveedor: 'aorus', producto: 'aorus socket am4 atx', noVentas: 1 },
  { proveedor: 'amd', producto: 'amd ryzen 3 1300', noVentas: 1 },
  { proveedor: 'amd', producto: 'amd ryzen 7 3800x', noVentas: 1 },
  { proveedor: 'amd', producto: 'amd ryzen 5 2500 ', noVentas: 1 },
  { proveedor: 'amd', producto: 'amd ryzen 9 5950x', noVentas: 1 },
  { proveedor: 'amd', producto: 'amd ryzen 7 2700x', noVentas: 1 },
  { proveedor: 'amd', producto: 'amd athlon', noVentas: 2 },
  { proveedor: 'amd', producto: 'amd ryzen 7 4550x', noVentas: 1 },
  { proveedor: 'amd', producto: 'amd ryzen 7 3700x', noVentas: 1 },
  { proveedor: 'amd', producto: 'amd ryzen 5 2500', noVentas: 1 },
  { proveedor: 'amd', producto: 'amd radeon 5700xt', noVentas: 1 },
  { proveedor: 'amd', producto: 'amd radeon 5500xt', noVentas: 1 },
  { proveedor: 'amd', producto: 'amd radeon rx590', noVentas: 2 }
]

```

```
Type "it" for more
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> it
[
  {
    proveedor: 'gigabyte',
    producto: 'gigabyte socket am4 atx',
    noVentas: 2
  },
  {
    proveedor: 'gigabyte',
    producto: 'gigabyte socket lga 1151 micro atx',
    noVentas: 1
  },
  {
    proveedor: 'gigabyte',
    producto: 'gigabyte socket am3 micro atx',
    noVentas: 1
  },
  {
    proveedor: 'hzxt',
    producto: 'hzxt socket am4 micro atx',
    noVentas: 1
  },
  {
    proveedor: 'hzxt',
    producto: 'hzxt socket lga 1151 micro atx',
    noVentas: 2
  }
]
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase>
```



La instalación del sistema operativo cuesta 150€ y queremos ver el precio de la máquina sin sistema operativo, para ello usamos variables y condicionales.

```
db.soldItems.aggregate( [
  {
    $project: {
      _id: 0,
      _idOrdenador:1,
      precio: 1,
      precioSinSistema: {
        $let: {
          vars: {
            price: {
              $cond: {
                if: '$caracteristicas.sistema_operativo',
                then: 150,
                else: 0
              }
            }
          },
          in: {
            $subtract: [
              "$precio",
              "$$price"
            ]
          }
        }
      }
    }
  }
] )
```

Declaramos una variable llamada price cuyo valor es un condicional, si sistema operativo es true es igual a 150, si es false es igual a 0, luego se le resta la variable \$\$price al precio de la máquina.

```
{ _idOrdenador: 11, precio: 900.6, precioSinSistema: 750.6 },
{ _idOrdenador: 12, precio: 1200.34, precioSinSistema: 1050.34 },
{ _idOrdenador: 13, precio: 500.5, precioSinSistema: 350.5 },
{ _idOrdenador: 21, precio: 800, precioSinSistema: 800 },
{ _idOrdenador: 22, precio: 750.4, precioSinSistema: 750.4 },
{ _idOrdenador: 23, precio: 2500.99, precioSinSistema: 2350.99 },
{ _idOrdenador: 31, precio: 1300.1, precioSinSistema: 1300.1 },
{ _idOrdenador: 41, precio: 800.1, precioSinSistema: 800.1 },
{ _idOrdenador: 42, precio: 500, precioSinSistema: 500 },
{ _idOrdenador: 43, precio: 5500, precioSinSistema: 5350 },
{ _idOrdenador: 44, precio: 800.89, precioSinSistema: 800.89 },
{ _idOrdenador: 51, precio: 1200.1, precioSinSistema: 1050.1 },
{ _idOrdenador: 52, precio: 2000.99, precioSinSistema: 2000.99 },
{ _idOrdenador: 61, precio: 1400.5, precioSinSistema: 1250.5 },
{ _idOrdenador: 71, precio: 2400.05, precioSinSistema: 2400.05 },
{ _idOrdenador: 72, precio: 800.99, precioSinSistema: 650.99 },
{ _idOrdenador: 73, precio: 3000.99, precioSinSistema: 2850.99 },
{ _idOrdenador: 81, precio: 700.5, precioSinSistema: 550.5 },
{ _idOrdenador: 91, precio: 1200.1, precioSinSistema: 1200.1 },
{ _idOrdenador: 92, precio: 3100.99, precioSinSistema: 2950.99 }
```

Queremos ver todo el dinero que ha generado cada vendedor ordenado de mayor a menor, teniendo en cuenta que hay que restarle el precio del sistema ya que es algo que hace nuestra empresa así que los vendedores no se llevan esa parte.

```
db.sales.aggregate( [
  {
    $lookup:{
      from: 'soldItems',
      localField: 'idVenta',
      foreignField: 'idVenta',
      as: 'equipo'
    }
  },
  {
    $unwind: {
      path:"$equipo"
    }
  },
  {
    $project: {
      _id: 0,
      vendedor:1,
      _idOrdenador:"$equipo._idOrdenador",
      precio: 1,
      ganancias: {
        $let: {
          vars: {
            price: {
              $cond: {
                if: '$equipo.caracteristicas.sistema_operativo',
                then: 150,
                else: 0
              }
            }
          },
          in: {
            $subtract: [
              "$equipo.precio",
              "$$price"
            ]
          }
        }
      }
    }
  },
  {
    $group:{
      _id: "$vendedor",
      ganancias:{
        $sum: "$ganancias"
      }
    }
  },
  {
    $sort:{
      ganancias:-1
    }
  }
] )
```

```
{ _id: 'Informatica Sur', ganancias: 13353.02 },
{ _id: 'Trastienda', ganancias: 6851.09 },
{ _id: 'Real Microinformatica', ganancias: 5202.53 },
{ _id: 'Tecnoshop', ganancias: 5151.889999999999 },
{ _id: 'Mediamark', ganancias: 1850.6 }

atlases atlas-s3z2gz-shard-0 [primary] myFirstDatabase>
```

Ver el mejor comprador, saber cuanto dinero ha gastado cada uno ordenado de mayor a menor.

```
db.sales.aggregate( [
  {
    $lookup: {
      from: 'soldItems',
      localField: 'idVenta',
      foreignField: 'idVenta',
      as: 'productos'
    }
  },
  {
    $project: {
      _id: 0,
      comprador: 1,
      precioCompras: "$productos.precio",
    }
  },
  {
    $unwind: {
      path: "$precioCompras"
    }
  },
  {
    $group: {
      _id: "$comprador",
      precioCompras: {
        $sum: "$precioCompras"
      }
    }
  },
  {
    $project: {
      _id: 1,
      precioCompras: {
        $round: [
          "$precioCompras",
          0
        ]
      },
    }
  },
  {
    $sort: {
      precioCompras: -1
    }
  }
] )
```

```
{ _id: 'Instituto Jesus del Granpoder', precioCompras: 8803 },
{ _id: 'Universidad Pablo de Olavide', precioCompras: 7601 },
{ _id: 'GMT', precioCompras: 7001 },
{ _id: 'TeyMa', precioCompras: 4051 },
{ _id: 'Almacenes La Isla', precioCompras: 3201 },
{ _id: 'Particular: Manuel Gutierrez', precioCompras: 1400 },
{ _id: 'Particular: Joaquin Jimenez', precioCompras: 1300 },
{ _id: 'Particular: Pablo Lopez', precioCompras: 700 }

clas atlas-s3z2gz-shard-0 [primary] myFirstDatabase>
```

Informatica Sur quiere hacer un descuento del 15% a los compradores que hayan comprado un ordenador de más de 1500€

```
db.sales.aggregate( [
  {
    $lookup: {
      from: 'soldItems',
      localField: 'idVenta',
      foreignField: 'idVenta',
      as: 'equipo'
    }
  },
  {
    $unwind: {
      path: "$equipo"
    }
  },
  {
    $match: {
      "vendedor": "Informatica Sur"
    }
  },
  {
    $project: {
      _id: 0,
      precio: "$equipo.precio",
      _idOrdenador: "$equipo._idOrdenador",
      precioDescuento: {
        $let: {
          vars: {
            descuento: {
              $cond: {
                if: {
                  $gt: [
                    "$equipo.precio",
                    1500
                  ]
                },
                then: 0.85,
                else: 1
              }
            }
          },
          in: {
            $multiply: [
              "$equipo.precio",
              "$$descuento"
            ]
          }
        }
      }
    }
  }
],
)
```

```

{
  $addField: {
    applyDiscount: {
      $cond: {
        if: {
          $gte: [
            "$precio",
            1500
          ]
        },
        then: true,
        else: false
      }
    }
  },
  {
    $project: {
      _idOrdenador: 1,
      applyDiscount: 1,
      price: "$precioDescuento"
    }
  },
  {
    $sort: {
      price: -1
    }
  }
}
] )

```

```

[
  { _idOrdenador: 43, applyDiscount: true, price: 4675 },
  { _idOrdenador: 73, applyDiscount: true, price: 2550.8414999999995 },
  { _idOrdenador: 71, applyDiscount: true, price: 2040.0425 },
  { _idOrdenador: 72, applyDiscount: false, price: 800.99 },
  { _idOrdenador: 44, applyDiscount: false, price: 800.89 },
  { _idOrdenador: 41, applyDiscount: false, price: 800.1 },
  { _idOrdenador: 42, applyDiscount: false, price: 500 }
]
Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase>

```

## 4. Uso de Mongoexport y Mongoimport.

Debemos descargar MongoDB Database Tools:

### MongoDB Database Tools


The MongoDB Database Tools are a collection of command-line utilities for working with a MongoDB deployment. These tools release independently from the MongoDB Server schedule enabling you to receive more frequent updates and leverage new features as soon as they are available. See the [MongoDB Database Tools](#) documentation for more information.

#### Available Downloads

Version  
100.5.2

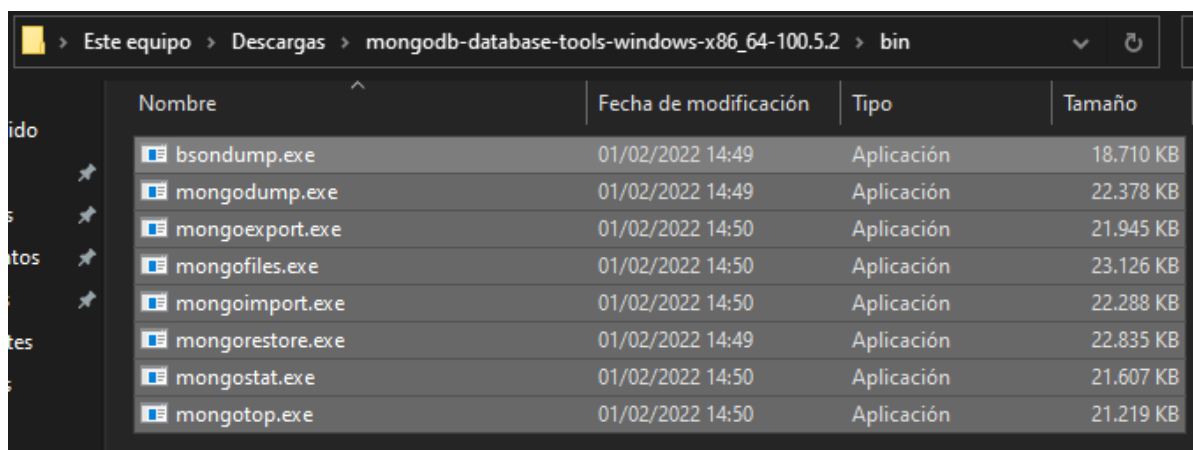
Platform  
Windows x86\_64

Package  
zip

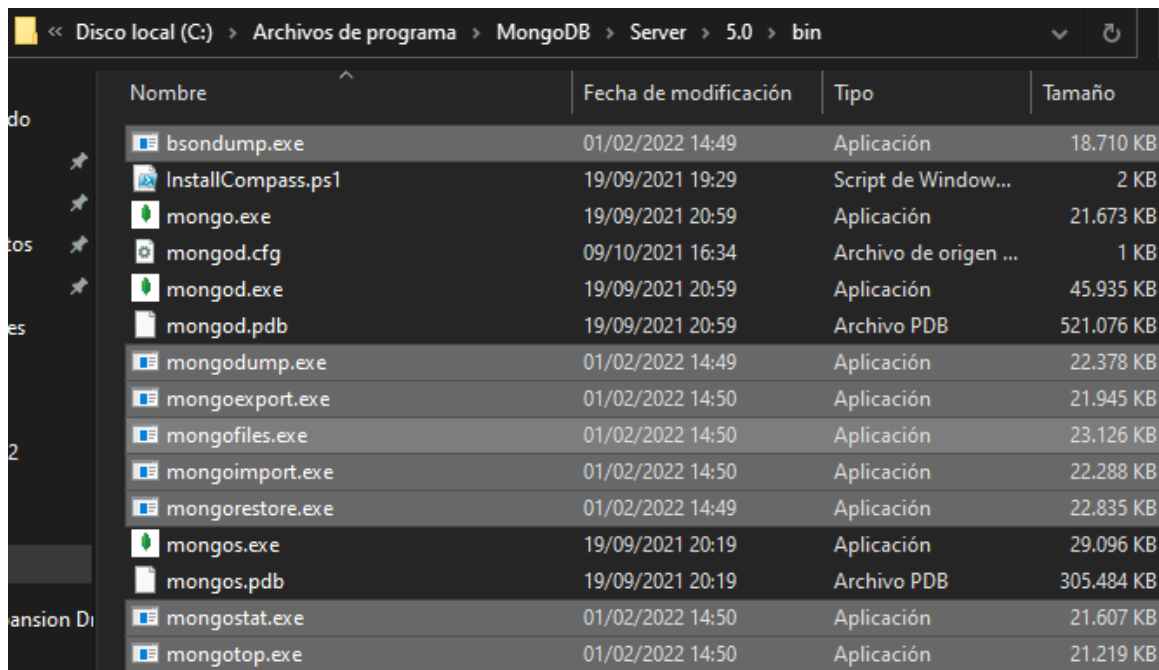
 **Download** [Copy Link](#)

[Documentation](#)  
[Archived releases](#)

Una vez descargado, movemos los ejecutables a la carpeta bin de MongoDB o bien lo añadimos al Path (yo lo guardo en la carpeta bin ya que es mas sencillo).



| Nombre           | Fecha de modificación | Tipo       | Tamaño    |
|------------------|-----------------------|------------|-----------|
| bsondump.exe     | 01/02/2022 14:49      | Aplicación | 18.710 KB |
| mongodump.exe    | 01/02/2022 14:49      | Aplicación | 22.378 KB |
| mongoexport.exe  | 01/02/2022 14:50      | Aplicación | 21.945 KB |
| mongofiles.exe   | 01/02/2022 14:50      | Aplicación | 23.126 KB |
| mongoimport.exe  | 01/02/2022 14:50      | Aplicación | 22.288 KB |
| mongorestore.exe | 01/02/2022 14:49      | Aplicación | 22.835 KB |
| mongostat.exe    | 01/02/2022 14:50      | Aplicación | 21.607 KB |
| mongotop.exe     | 01/02/2022 14:50      | Aplicación | 21.219 KB |



| Nombre             | Fecha de modificación | Tipo                  | Tamaño     |
|--------------------|-----------------------|-----------------------|------------|
| bsondump.exe       | 01/02/2022 14:49      | Aplicación            | 18.710 KB  |
| InstallCompass.ps1 | 19/09/2021 19:29      | Script de Window...   | 2 KB       |
| mongo.exe          | 19/09/2021 20:59      | Aplicación            | 21.673 KB  |
| mongod.cfg         | 09/10/2021 16:34      | Archivo de origen ... | 1 KB       |
| mongod.exe         | 19/09/2021 20:59      | Aplicación            | 45.935 KB  |
| mongod.pdb         | 19/09/2021 20:59      | Archivo PDB           | 521.076 KB |
| mongodump.exe      | 01/02/2022 14:49      | Aplicación            | 22.378 KB  |
| mongoexport.exe    | 01/02/2022 14:50      | Aplicación            | 21.945 KB  |
| mongofiles.exe     | 01/02/2022 14:50      | Aplicación            | 23.126 KB  |
| mongoimport.exe    | 01/02/2022 14:50      | Aplicación            | 22.288 KB  |
| mongorestore.exe   | 01/02/2022 14:49      | Aplicación            | 22.835 KB  |
| mongos.exe         | 19/09/2021 20:19      | Aplicación            | 29.096 KB  |
| mongos.pdb         | 19/09/2021 20:19      | Archivo PDB           | 305.484 KB |
| mongostat.exe      | 01/02/2022 14:50      | Aplicación            | 21.607 KB  |
| mongotop.exe       | 01/02/2022 14:50      | Aplicación            | 21.219 KB  |

## 4.1.Mongoexport.

Para realizar mongo export usamos la siguiente sintaxis:

```
PS C:\Users\grans> mongoexport --uri="mongodb+srv://gonzalo:lioco10@cluster0.segc7.mongodb.net/myFirstDatabase" --collection=soldItems --out=soldItems.json
2022-03-07T01:12:14.047+0100 connected to: mongodb+srv://[**REDACTED**]@cluster0.segc7.mongodb.net/myFirstDatabase
2022-03-07T01:12:14.335+0100 exported 21 records
PS C:\Users\grans>
```

- uri → enlace de Mongo Atlas
- collection → colección deseada
- out → fichero de salida
- db → base de datos deseada

Ejemplo de uso en local:

```
PS C:\Users\grans> mongoexport --collection=soldItems --db=VentasOrdenadores --out=soldItems.json
2022-03-06T16:13:20.888+0100 connected to: mongodb://localhost/
2022-03-06T16:13:20.973+0100 exported 21 records
PS C:\Users\grans> mongoexport --collection=sales --db=VentasOrdenadores --out=sales.json
2022-03-06T16:15:04.902+0100 connected to: mongodb://localhost/
2022-03-06T16:15:04.941+0100 exported 9 records
PS C:\Users\grans> mongoexport --collection=providers --db=VentasOrdenadores --out=providers.json
2022-03-06T16:15:19.272+0100 connected to: mongodb://localhost/
2022-03-06T16:15:19.314+0100 exported 9 records
PS C:\Users\grans> mongoexport --collection=products --db=VentasOrdenadores --out=products.json
2022-03-06T16:15:39.645+0100 connected to: mongodb://localhost/
2022-03-06T16:15:39.683+0100 exported 65 records
PS C:\Users\grans>
```

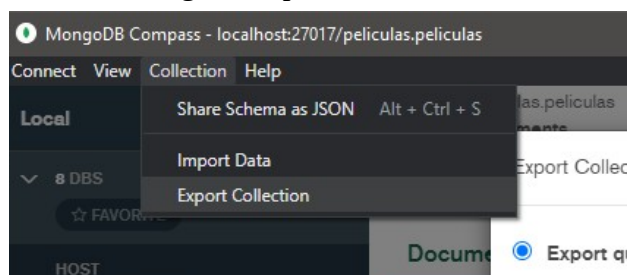
Uso en Mongo Atlas:

```

Atlas atlas-s3z2gz-shard-0 [primary] myFirstDatabase> exit
PS C:\Users\grans> mongoexport --uri="mongodb+srv://gonzalo:lioco10@cluster0.segc7.mongodb.net/myFirstDatabase" --collection=soldItems --out=soldItems.json
2022-03-07T01:12:14.047+0100 connected to: mongodb+srv://[**REDACTED**]@cluster0.segc7.mongodb.net/myFirstDatabase
2022-03-07T01:12:14.335+0100 exported 21 records
PS C:\Users\grans> mongoexport --uri="mongodb+srv://gonzalo:lioco10@cluster0.segc7.mongodb.net/myFirstDatabase" --collection=sales --out=sales.json
2022-03-07T01:15:15.590+0100 connected to: mongodb+srv://[**REDACTED**]@cluster0.segc7.mongodb.net/myFirstDatabase
2022-03-07T01:15:15.868+0100 exported 9 records
PS C:\Users\grans> mongoexport --uri="mongodb+srv://gonzalo:lioco10@cluster0.segc7.mongodb.net/myFirstDatabase" --collection=providers --out=providers.json
2022-03-07T01:15:31.424+0100 connected to: mongodb+srv://[**REDACTED**]@cluster0.segc7.mongodb.net/myFirstDatabase
2022-03-07T01:15:31.730+0100 exported 9 records
PS C:\Users\grans> mongoexport --uri="mongodb+srv://gonzalo:lioco10@cluster0.segc7.mongodb.net/myFirstDatabase" --collection=products --out=products.json
2022-03-07T01:15:48.036+0100 connected to: mongodb+srv://[**REDACTED**]@cluster0.segc7.mongodb.net/myFirstDatabase
2022-03-07T01:15:48.341+0100 exported 65 records
PS C:\Users\grans>

```

### Para Mongo Compass



Pulsamos en esa opción y seleccionamos los campos que queremos.

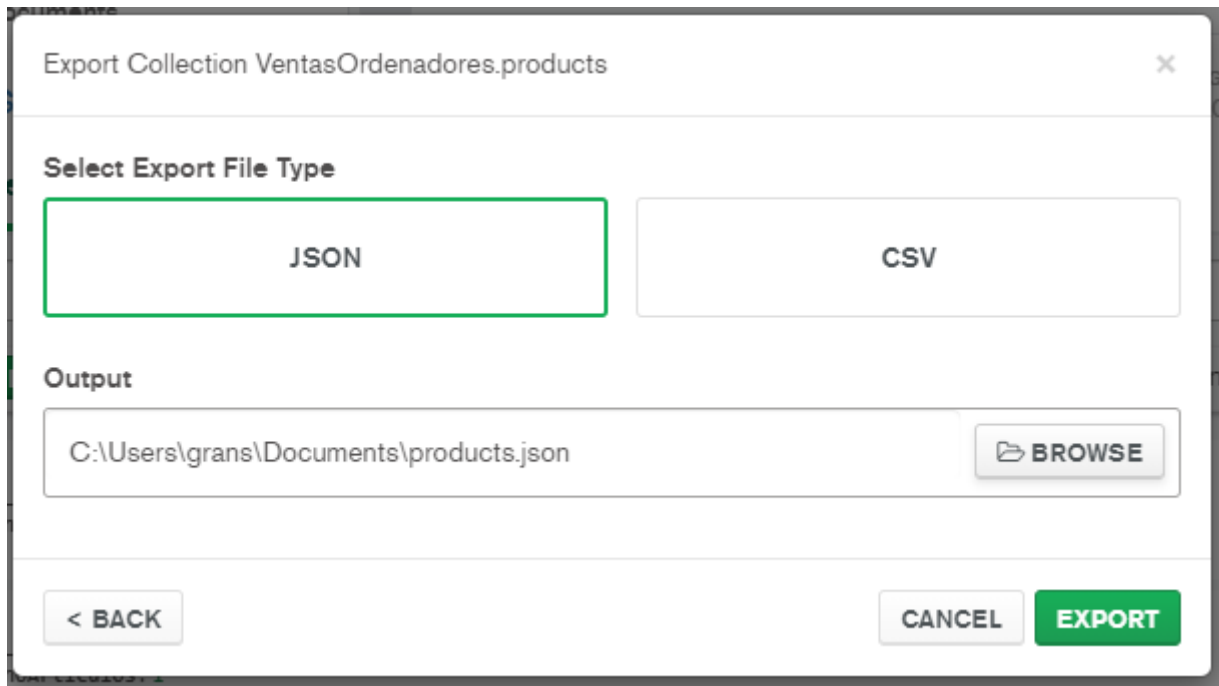
The screenshot shows the 'Select Fields' dialog box. It has a title bar with 'Select Fields' and an information icon. Below the title is a table with columns for selection checkboxes, an index, and the field name. The first three fields are selected with blue checkmarks.

|                                     |   | Field Name  |
|-------------------------------------|---|-------------|
| <input checked="" type="checkbox"/> |   | Field Name  |
| <input checked="" type="checkbox"/> | 1 | _id.idMarca |
| <input checked="" type="checkbox"/> | 2 | _id.name    |
| <input checked="" type="checkbox"/> | 3 | noArticulos |
| <input type="checkbox"/>            | 4 | Add field   |

At the bottom right of the table, there is a text 'to add' with a double arrow icon. Below the table are three buttons: '< BACK' (disabled), 'CANCEL' (disabled), and 'SELECT OUTPUT' (active/green).

Después seleccionamos un directorio y el tipo de fichero donde lo queremos guardar.





## 4.2.Mongoimport

La sintaxis de mongoimport es la siguiente:

- db → base de datos
- collection → nombre colección
- file → archivo a importar
- uri → enlace de Mongo Atlas

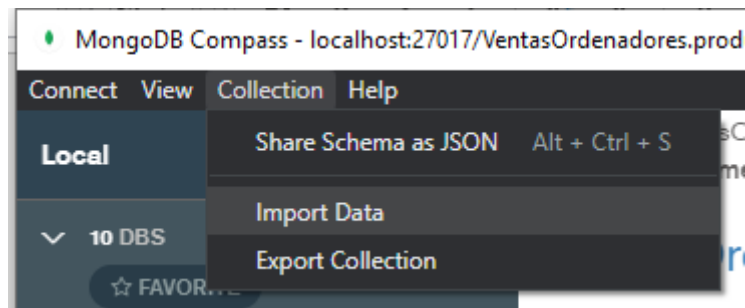
Ejemplo de mongoimport en local:

```
PS C:\Users\grans> mongoimport --db=Proyecto02Trimestre --collection=soldItems --file=soldItems.json
2022-03-07T01:25:42.870+0100 connected to: mongodb://localhost/
2022-03-07T01:25:43.029+0100 21 document(s) imported successfully. 0 document(s) failed to import.
PS C:\Users\grans> mongoimport --db=Proyecto02Trimestre --collection=sales --file=sales.json
2022-03-07T01:26:06.757+0100 connected to: mongodb://localhost/
2022-03-07T01:26:06.905+0100 9 document(s) imported successfully. 0 document(s) failed to import.
PS C:\Users\grans> mongoimport --db=Proyecto02Trimestre --collection=products --file=products.json
2022-03-07T01:26:27.668+0100 connected to: mongodb://localhost/
2022-03-07T01:26:27.812+0100 65 document(s) imported successfully. 0 document(s) failed to import.
PS C:\Users\grans> mongoimport --db=Proyecto02Trimestre --collection=providers --file=providers.json
2022-03-07T01:26:41.001+0100 connected to: mongodb://localhost/
2022-03-07T01:26:41.145+0100 9 document(s) imported successfully. 0 document(s) failed to import.
PS C:\Users\grans>
```

Ejemplo de mongoimport en Atlas:

```
PS C:\Users\grans> mongoimport --uri "mongodb+srv://gonzalo:lioco10@cluster0.segc7.mongodb.net/dbVentas" --collection=soldItems --file=soldItems.json
2022-03-07T01:30:49.604+0100 connected to: mongodb+srv://[**REDACTED**]@cluster0.segc7.mongodb.net/dbVentas
2022-03-07T01:30:49.769+0100 21 document(s) imported successfully. 0 document(s) failed to import.
PS C:\Users\grans>
```

Ejemplo de mongoimport en Compass:



Seleccionamos la opción importar, seleccionamos json y el archivo a importar.

