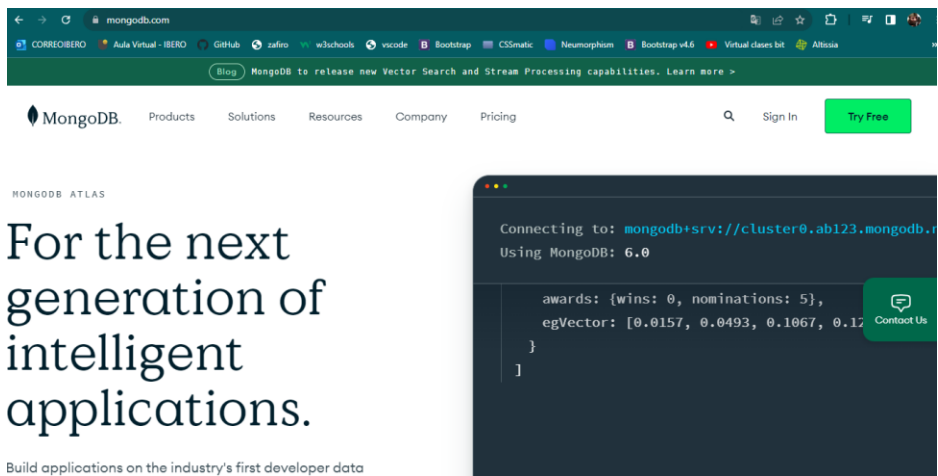
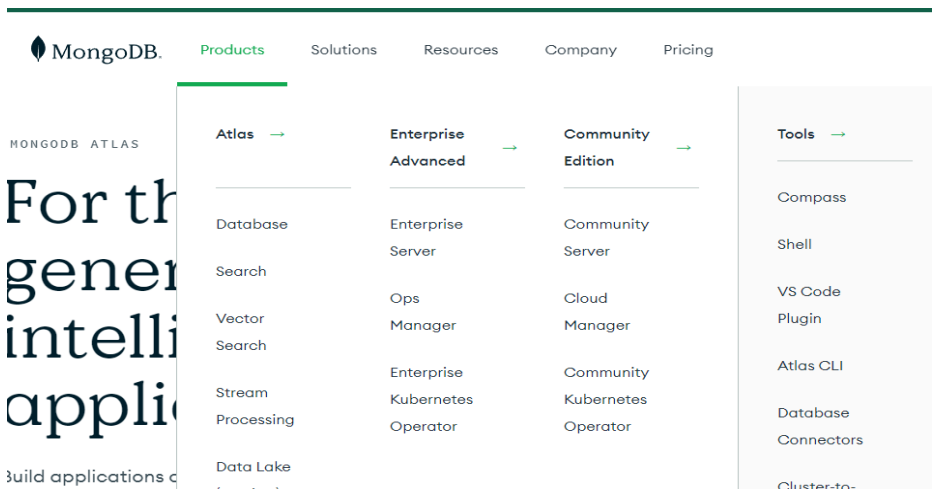


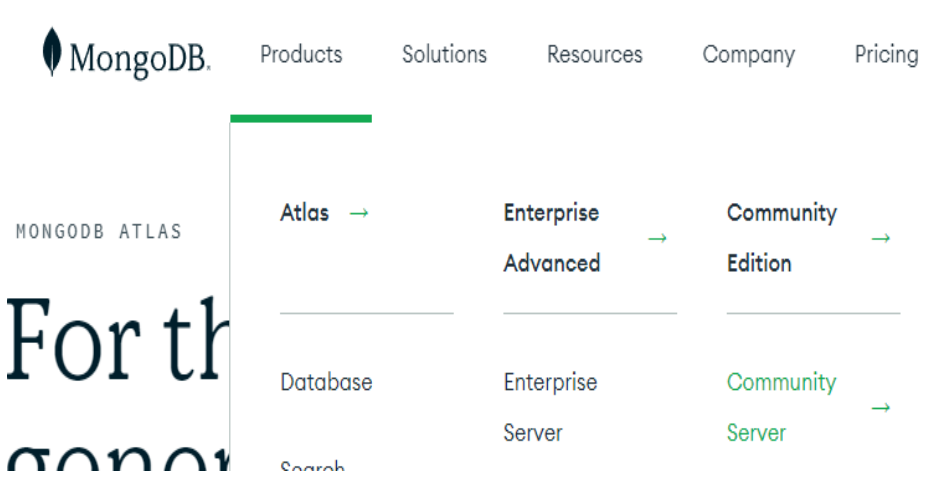
Primero entramos a mongodb.com



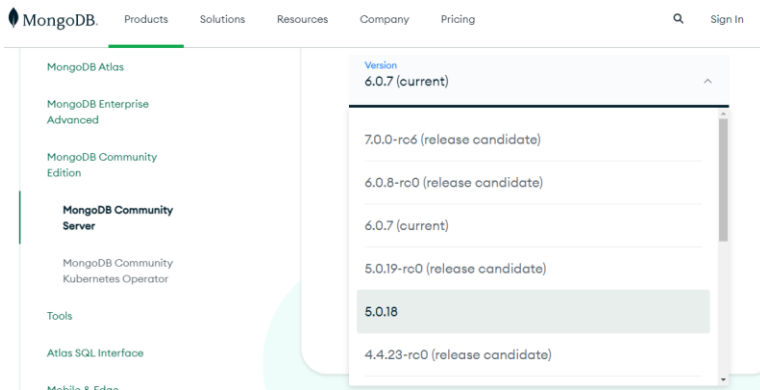
Dentro de la pagina de Mongodb nos dirigiremos a Productos



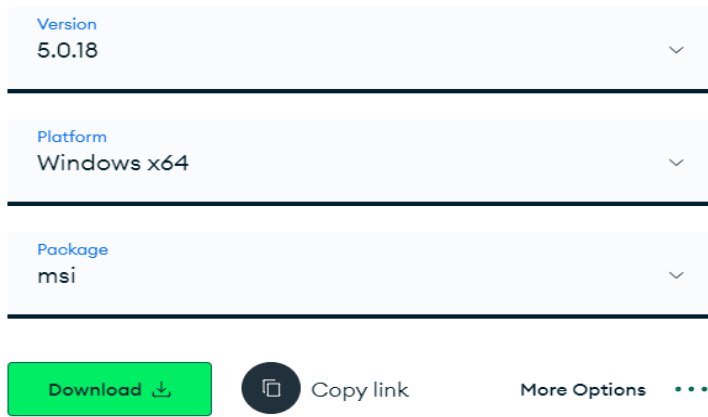
Dentro de Productos vamos a ingresar a community server



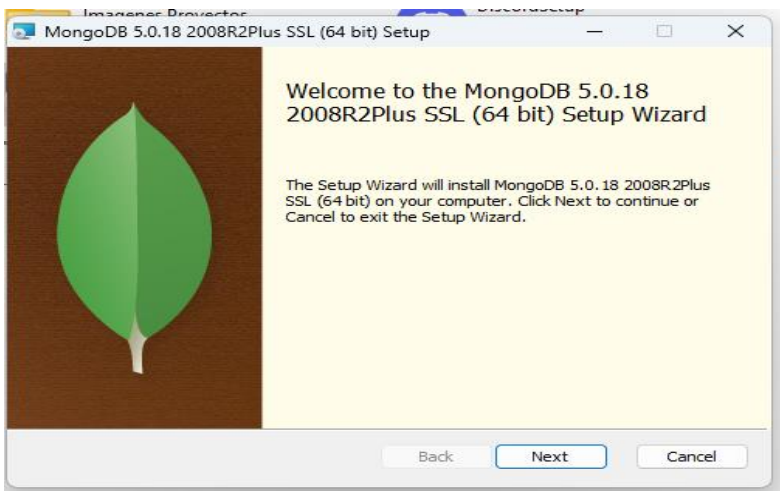
Dentro de community server podemos descargar la versión de mongo que deseamos utilizar, en este caso usaremos la versión 5.0.18 para Windows de 64 bits



Una vez seleccionada la versión damos clic en descargar

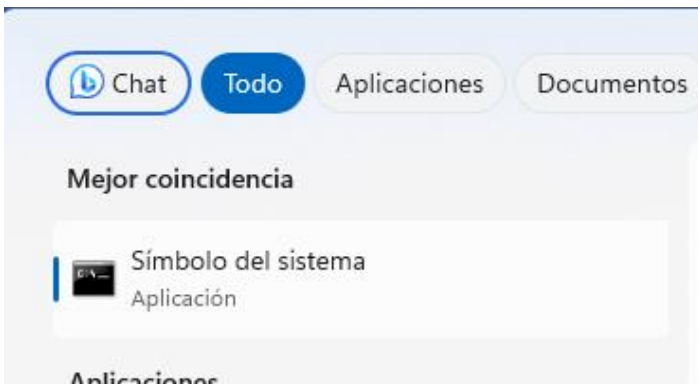


Una vez descargado procederemos a instalar en nuestro equipo



Puede aparecer que mongo es un virus troyano pero vamos a omitir y decirle que es seguro ya que este trabaja por medio de puertos

Una vez instalado vamos a abrir un cmd o símbolo de sistema



Una vez estemos en símbolo de sistema o CMD vamos a posicionarnos en la ruta donde dejamos instalado mongo

En este caso la instalamos en la unidad D, para acceder colocamos D:, esto nos llevara a la unidad D.

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22621.1928]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Dell>d:

D:\>
```

Ahora colocaremos mongo DB, para visualizar que este instalado y validar que versión tenemos

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22621.1928]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Dell>D:

D:\>mongo DB
```

Damos Enter y nos mostrara la versión instalada la cual debe corresponder con la que seleccionamos en este caso como podemos apreciar instalamos la v5.0.18

```
Símbolo del sistema - mongo
Microsoft Windows [Versión 10.0.22621.1928]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Dell>D:

D:\>mongo DB
MongoDB shell version v5.0.18
connecting to: mongodb://127.0.0.1:27017/DB?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("e83c8744-11a8-4fa1-8039-164dcb69dbc4") }
MongoDB server version: 5.0.18
=====
```

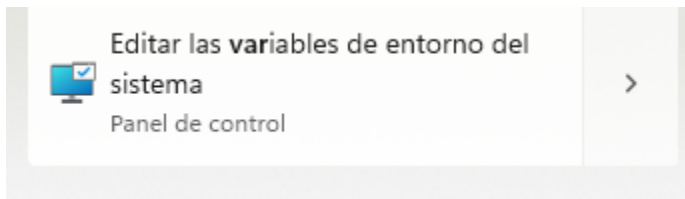
Ahora validamos el campo que indica connecting si este campo no nos sale entonces deberemos realizar una configuración adicional

```
Símbolo del sistema - mongo
Microsoft Windows [Versión 10.0.22621.1928]
(c) Microsoft Corporation. Todos los derechos reservados.

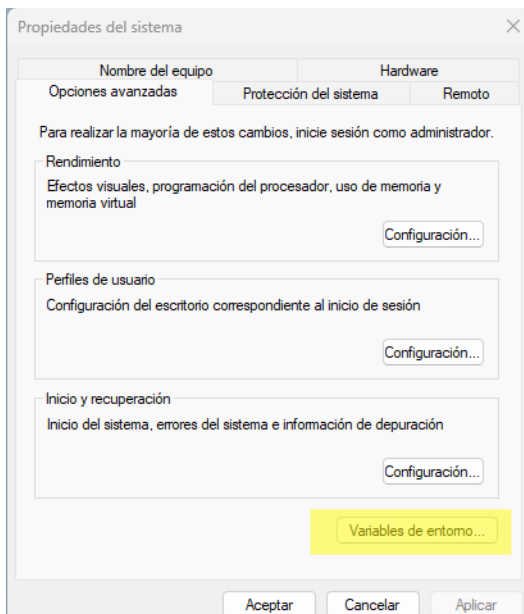
C:\Users\Dell>D:

D:\>mongo DB
MongoDB shell version v5.0.18
connecting to: mongodb://127.0.0.1:27017/DB?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("e83c8744-11a8-4fa1-8039-164dcb69dbc4") }
MongoDB server version: 5.0.18
=====
```

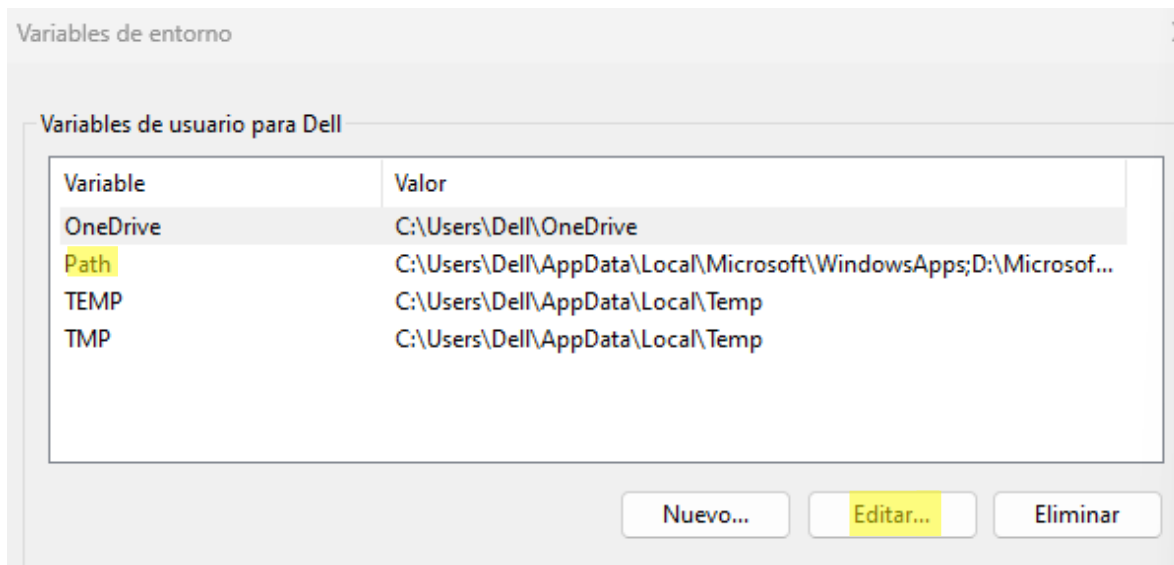
Para ello escribiremos en la barra de búsqueda de inicio lo siguiente: variables de entorno



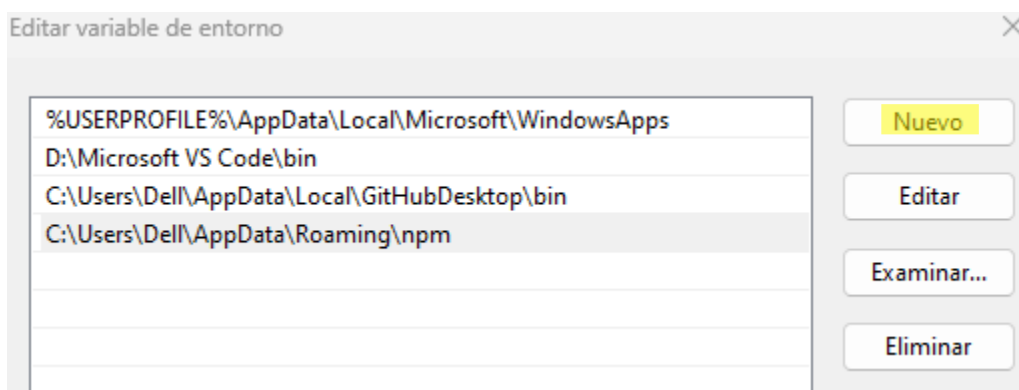
Una vez ingresamos a variables de entorno nos dirigimos a la etiqueta variable de entorno



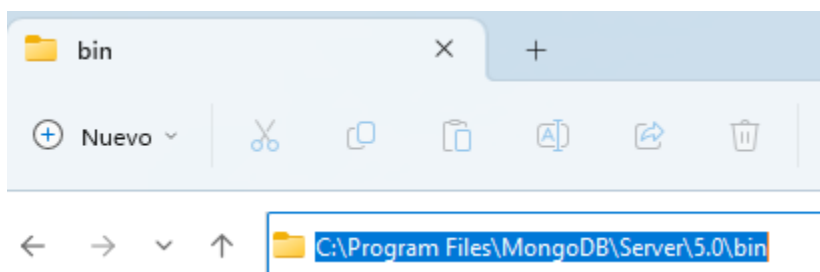
En la ventana que nos aparece vamos a ubicarnos sobre path luego vamos a darle editar



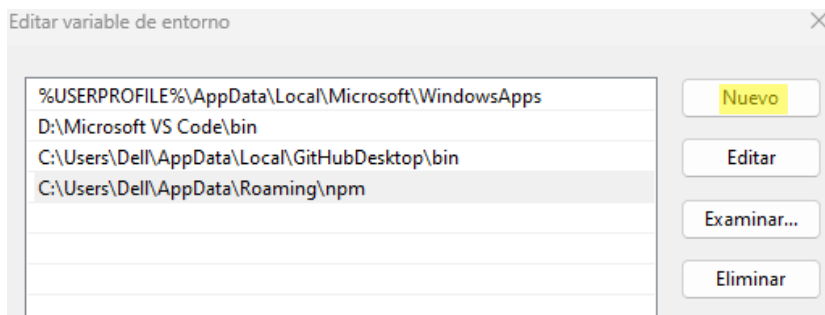
Nos aparecerá una ventana donde debemos ingresar la ruta donde quedo instalado el .bin de mongo



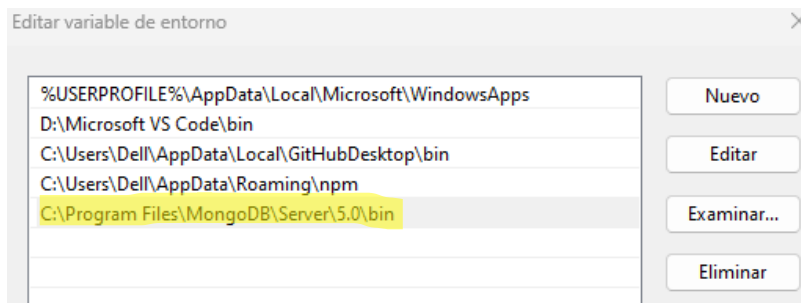
Para poder crear la ruta del path debemos buscar en la carpeta donde se instalo mongo y buscar la ruta que nos lleva al .bin, en este caso quedo en la ruta de la imagen, la copiamos y vamos a la imagen anterior



En esta vamos a dar clic en el botón nuevo, aparecerá un recuadro donde colocaremos la dirección copiada con anterioridad



Nos quedara algo así



Damos aceptar luego aceptar y posteriormente aceptar con esto ya podremos visualizar la versión y que tenemos conexión con mongo en un direccionamiento local

```
Símbolo del sistema - mongo X + v
Microsoft Windows [Versión 10.0.22621.1928]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Dell>D:

D:\>mongo DB
MongoDB shell version v5.0.18
connecting to: mongodb://127.0.0.1:27017/DB?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("e83c8744-11a8-4fa1-8039-164dcb69dbc4") }
MongoDB server version: 5.0.18
=====
```

Con show dbs, siendo la abreviatura de show databases, con los cuales visualizamos las bases de datos que tenemos creadas en nuestro equipo

```
---
> show dbs
admin    0.000GB
config  0.000GB
local    0.000GB
> |

---
> show databases
admin    0.000GB
config  0.000GB
local    0.000GB
> |
```

Con use creamos o accedemos a la base de datos, en este caso como no se tenia creada la base de datos creamos la base con el nombre de Actividad_5, y la buscamos con show databases

```
The server generated these status messages:
2023-07-01T09:08:48.483Z
configuration is unrestricted
---
> show dbs
admin    0.000GB
config   0.000GB
local    0.000GB
> use Actividad_5
switched to db Actividad_5
> |

---
> show databases
Actividad_5  0.000GB
admin         0.000GB
config        0.000GB
local         0.000GB
> |
```

Ahora ingresamos a la base de datos usando use Actividad_5 siendo que ya se encuentra creada nos saldrá de la siguiente manera

```
> show databases
Actividad_5  0.000GB
admin         0.000GB
config        0.000GB
local         0.000GB
> use Actividad_5
switched to db Actividad_5
> |
```

Vamos a crear una colección en este caso la vamos a llamar Familia, existen dos maneras de crearla, en este caso la crearemos con db.createCollection("Familia")

```
> show dbs
admin    0.000GB
config   0.000GB
local    0.000GB
> use Actividad_5
switched to db Actividad_5
> db.createCollection("Familia")
{ "ok" : 1 }
> |
```

Vamos a crear una colección en este caso la vamos a llamar Familiares, pero lo vamos a hacer con insertando un elemento en formato json es decir db.Familiares.insertOne({nombre:"Angel"})

```
> show databases
Actividad_5  0.000GB
admin        0.000GB
config       0.000GB
local        0.000GB
> use Actividad_5
switched to db Actividad_5
> db.Familiares.insertOne({nombre:"Angel"})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("64a091ef5330f85995f18a4d")
}
> |
```

De esta manera podemos visualizar que se crea la colección de Familiares, al igual que estamos viendo que contiene la base de datos que creamos, para ello usamos el comando show collections

```
> show databases
Actividad_5  0.000GB
admin        0.000GB
config       0.000GB
local        0.000GB
> use Actividad_5
switched to db Actividad_5
> show collections
Familia
Familiares
> |
```

Ahora vamos a ingresar a la colección de Familia, para ello usamos el comando use Familia la cual nos entrega un switched to db Familia el cual nos indica que estamos dentro de la colección

```
> use Familia
switched to db Familia
> |
```


Vamos a crear un registro con los campos nombre, apellido, dirección, teléfono, edad, cedula, si el resultado es exitoso nos indicara que el usuario quedo registrado con la palabra true, adicional nos indicara un id único para el usuario, como lo muestra en la imagen a continuación.

```
db.Familia.insertOne({nombre:"Angel",apellido:"Rangel",edad:5,cedula:10000000000,telefono:6070000000,direccion:"condado campestre"})
```

```
> use Actividad_5
switched to db Actividad_5
> show collections
Familia
Familiares
> db.Familia.insertOne({nombre:"Angel",apellido:"Rangel",edad:5,cedula:10000000000,telefono:6070000000,direccion:"condado campestre"})
{acknowledged: true, insertedId: ObjectId("64a0960b686b90fa2164e952")}
```

Con el comando db.nombre de la colección.find() podemos validar los datos que están registrados dentro de una colección

```
> db.Familia.find()
{ "_id" : ObjectId("64a0960b686b90fa2164e952"), "nombre" : "Angel", "apellido" : "Rangel", "edad" : 5, "cedula" : 10000000000, "telefono" : 6070000000, "direccion" : "condado campestre" }
```

```
> show dbs
Actividad_5 0.000GB
Familia      0.000GB
admin        0.000GB
config       0.000GB
local        0.000GB
> use Actividad_5
switched to db Actividad_5
> show collections
Familia
Familiares
> db.Familiares.find()
{ "_id" : ObjectId("64a091ef5330f85995f18a4d"), "nombre" : "Angel" }
> db.Familia.find()
{ "_id" : ObjectId("64a0960b686b90fa2164e952"), "nombre" : "Angel", "apellido" : "Rangel", "edad" : 5, "cedula" : 10000000000, "telefono" : 6070000000, "direccion" : "condado campestre" }
> use Familia
switched to db Familia
> db.Familia.find()
{ "_id" : ObjectId("64a0960b686b90fa2164e952"), "nombre" : "Angel", "apellido" : "Rangel", "edad" : 5, "cedula" : 10000000000, "telefono" : 6070000000, "direccion" : "condado campestre" }
```

Ahora bien para poderlo visualizar mejor podemos usar el comando: db.nombredecoleccion.find().pretty() que nos ayuda a listar en forma vertical para tener una mejor visual de los datos

```
> db.Familia.find().pretty()
{
  "_id" : ObjectId("64a0960b686b90fa2164e952"),
  "nombre" : "Angel",
  "apellido" : "Rangel",
  "edad" : 5,
  "cedula" : 10000000000,
  "telefono" : 6070000000,
  "direccion" : "condado campestre"
}
```

Podemos también ingresar barrios registros usando

`db.nombredelacoleccion.insertMany({clave:"llave1"},{clave:"llave2"}))`, una vez ingresados nos quedara la comprobación como se muestra a continuación, donde ingresamos varios elementos con varios parametros

```
> db.Familia.insertMany([{"nombre":"Violeta",apellido:"Rangel",edad:3,cedula:1000000001,telefono:6070000001,direccion:"condado campestre"}, {"nombre":"Diana",apellido:"Osorio",edad:37,cedula:1000000002,telefono:6070000002,direccion:"condado campestre"}, {"nombre":"Fabiol",apellido:"Osoriol",edad:65,cedula:1000000003,telefono:6070000003,direccion:"condado campestre"}, {"nombre":"Emilce",apellido:"Cote",edad:60,cedula:1000000004,telefono:6070000004,direccion:"condado campestre"}, {"nombre":"Betsy",apellido:"Cote",edad:58,cedula:1000000005,telefono:6070000005,direccion:"condado campestre"}, {"nombre":"Daniel",apellido:"Cote",edad:34,cedula:1000000006,telefono:6070000006,direccion:"condado campestre"}])
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("64a0a51b9bf47055b4bce64a"),
    ObjectId("64a0a51b9bf47055b4bce64b"),
    ObjectId("64a0a51b9bf47055b4bce64c"),
    ObjectId("64a0a51b9bf47055b4bce64d"),
    ObjectId("64a0a51b9bf47055b4bce64e"),
    ObjectId("64a0a51b9bf47055b4bce64f")
  ]
}
```

Observando como quedan los registros vamos a visualizar solo id, nombre y cedula, esto lo hacemos con las propiedades de proyección que tiene `find()`, aplicamos el comando

`db.Familia.find({}, {apellido:0,edad:0,telefono:0,direccion:0}).pretty()`, donde 0 nos indica que ocultamos el campo

```
> db.Familia.find({}, {apellido:0,edad:0,telefono:0,direccion:0}).pretty()
{
  "_id" : ObjectId("64a0960b686b90fa2164e952"),
  "nombre" : "Angel",
  "cedula" : 100000000000
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64a"),
  "nombre" : "Violeta",
  "cedula" : 10000000001
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64b"),
  "nombre" : "Diana",
  "cedula" : 10000000002
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64c"),
  "nombre" : "Fabiol",
  "cedula" : 10000000003
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64d"),
  "nombre" : "Emilce",
  "cedula" : 10000000004
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64e"),
  "nombre" : "Betsy",
  "cedula" : 10000000005
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64f"),
  "nombre" : "Daniel",
  "cedula" : 10000000006
}
```

Para comparar la información ingresada vamos a dejarlo sin la proyección, el comando quedaría `db.Familia.find().pretty()`

```
> db.Familia.find().pretty()
{
  "_id" : ObjectId("64a0960b686b90fa2164e952"),
  "nombre" : "Angel",
  "apellido" : "Rangel",
  "edad" : 5,
  "cedula" : 1000000000,
  "telefono" : 6070000000,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64a"),
  "nombre" : "Violeta",
  "apellido" : "Rangel",
  "edad" : 3,
  "cedula" : 1000000001,
  "telefono" : 6070000001,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64b"),
  "nombre" : "Diana",
  "apellido" : "Osorio",
  "edad" : 37,
  "cedula" : 1000000002,
  "telefono" : 6070000002,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64c"),
  "nombre" : "Fabiol",
  "apellido" : "Osoriol",
  "edad" : 65,
  "cedula" : 1000000003,
  "telefono" : 6070000003,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64d"),
  "nombre" : "Emilce",
  "apellido" : "Cote",
  "edad" : 60,
  "cedula" : 1000000004,
  "telefono" : 6070000004,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64e"),
  "nombre" : "Betsy",
  "apellido" : "Cote",
  "edad" : 58,
  "cedula" : 1000000005,
  "telefono" : 6070000005,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64f"),
  "nombre" : "Daniel",
  "apellido" : "Cote",
  "edad" : 34,
  "cedula" : 1000000006,
  "telefono" : 6070000006,
  "direccion" : "condado campestre"
}
```

Resulta que la edad de Fabiol nos quedo mal entonces tenemos que corregirla o actualizarla para ello usamos el comando `db.collection.update({criterio de búsqueda},{ $set:{criterio a modificar:modificación}}`, en ejemplo `db.Familia.updateOne({nombre:"Fabiol"},{$set:{edad:75}})`, como vemos cambiamos la edad de 65 a 75

```
> db.Familia.updateOne({nombre:"Fabiol"},{$set:{edad:75}})
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.Familia.find().pretty()
{
  "_id" : ObjectId("64a0960b686b90fa2164e952"),
  "nombre" : "Angel",
  "apellido" : "Rangel",
  "edad" : 5,
  "cedula" : 1000000000,
  "telefono" : 6070000000,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64a"),
  "nombre" : "Violeta",
  "apellido" : "Rangel",
  "edad" : 3,
  "cedula" : 1000000001,
  "telefono" : 6070000001,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64b"),
  "nombre" : "Diana",
  "apellido" : "Osorio",
  "edad" : 37,
  "cedula" : 1000000002,
  "telefono" : 6070000002,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64c"),
  "nombre" : "Fabiol",
  "apellido" : "Osoriol",
  "edad" : 75,
  "cedula" : 1000000003,
  "telefono" : 6070000003,
  "direccion" : "condado campestre"
}
```

Ahora vamos a eliminar la primera persona registrada esto lo hacemos con el comando `db.coleccion.deleteOne({criterio:el criteriomanejado})`, vamos a usar el siguiente `db.Familia.deleteOne({nombre:"Angel"})`, como apreciamos en la imagen nos indica que fue eliminado el primer registro y si comparamos vemos que tenemos 6 registros

```
> db.Familia.deleteOne({nombre:"Angel"})
{ "acknowledged" : true, "deletedCount" : 1 }
> db.Familia.find().pretty()
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64a"),
  "nombre" : "Violeta",
  "apellido" : "Rangel",
  "edad" : 3,
  "cedula" : 1000000001,
  "telefono" : 6070000001,
  "direccion" : "condado campestre"
}
```

```
> db.Familia.find().pretty()
{
  "_id" : ObjectId("64a0960b686b90fa2164e952"),
  "nombre" : "Angel",
  "apellido" : "Rangel",
  "edad" : 5,
  "cedula" : 1000000000,
  "telefono" : 6070000000,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64a"),
  "nombre" : "Violeta",
  "apellido" : "Rangel",
  "edad" : 3,
  "cedula" : 1000000001,
  "telefono" : 6070000001,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64b"),
  "nombre" : "Diana",
  "apellido" : "Osorio",
  "edad" : 37,
  "cedula" : 1000000002,
  "telefono" : 6070000002,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64c"),
  "nombre" : "Fabiola",
  "apellido" : "Osorio",
  "edad" : 65,
  "cedula" : 1000000003,
  "telefono" : 6070000003,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64d"),
  "nombre" : "Emilce",
  "apellido" : "Cote",
  "edad" : 60,
  "cedula" : 1000000004,
  "telefono" : 6070000004,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64e"),
  "nombre" : "Betsy",
  "apellido" : "Cote",
  "edad" : 58,
  "cedula" : 1000000005,
  "telefono" : 6070000005,
  "direccion" : "condado campestre"
}
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64f"),
  "nombre" : "Daniel",
  "apellido" : "Cote",
  "edad" : 34,
  "cedula" : 1000000006,
  "telefono" : 6070000006,
  "direccion" : "condado campestre"
}
```

Visualizamos la cantidad de registros que tenemos con el comando `count()`, ejemplo `db.Familia.find().count()`

```
> db.Familia.find().count()
6
> |
```

Ahora colocaremos en orden por edad los registros, eso lo hacemos con el comando `db.colecion.find().sort({edad:1}).pretty()` donde el 1 que esta en edad nos condiciona para decir que es de menor a mayor, si colocamos -1 decimos que es de mayor a menor

```
> db.Familia.find().sort({edad:1}).pretty()
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64a"),
  "nombre" : "Violeta",
  "apellido" : "Rangel",
  "edad" : 3,
  "cedula" : 1000000001,
  "telefono" : 6070000001,
  "direccion" : "condado campestre"
}

{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64f"),
  "nombre" : "Daniel",
  "apellido" : "Cote",
  "edad" : 34,
  "cedula" : 1000000006,
  "telefono" : 6070000006,
  "direccion" : "condado campestre"
}

{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64b"),
  "nombre" : "Diana",
  "apellido" : "Osorio",
  "edad" : 37,
  "cedula" : 1000000002,
  "telefono" : 6070000002,
  "direccion" : "condado campestre"
}

{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64e"),
  "nombre" : "Betsy",
  "apellido" : "Cote",
  "edad" : 58,
  "cedula" : 1000000005,
  "telefono" : 6070000005,
  "direccion" : "condado campestre"
}

{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64d"),
  "nombre" : "Emilce",
  "apellido" : "Cote",
  "edad" : 60,
  "cedula" : 1000000004,
  "telefono" : 6070000004,
  "direccion" : "condado campestre"
}

{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64c"),
  "nombre" : "Fabiol",
  "apellido" : "Osoriol",
  "edad" : 75,
  "cedula" : 1000000003,
  "telefono" : 6070000003,
  "direccion" : "condado campestre"
}
```

Ahora lo aremos con mayor a menor

```
> db.Familia.find().sort({edad:-1}).pretty()
{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64c"),
  "nombre" : "Fabiol",
  "apellido" : "Osoriol",
  "edad" : 75,
  "cedula" : 1000000003,
  "telefono" : 6070000003,
  "direccion" : "condado campestre"
}

{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64d"),
  "nombre" : "Emilce",
  "apellido" : "Cote",
  "edad" : 60,
  "cedula" : 1000000004,
  "telefono" : 6070000004,
  "direccion" : "condado campestre"
}

{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64e"),
  "nombre" : "Betsy",
  "apellido" : "Cote",
  "edad" : 58,
  "cedula" : 1000000005,
  "telefono" : 6070000005,
  "direccion" : "condado campestre"
}

{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64b"),
  "nombre" : "Diana",
  "apellido" : "Osorio",
  "edad" : 37,
  "cedula" : 1000000002,
  "telefono" : 6070000002,
  "direccion" : "condado campestre"
}

{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64f"),
  "nombre" : "Daniel",
  "apellido" : "Cote",
  "edad" : 34,
  "cedula" : 1000000006,
  "telefono" : 6070000006,
  "direccion" : "condado campestre"
}

{
  "_id" : ObjectId("64a0a51b9bf47055b4bce64a"),
  "nombre" : "Violeta",
  "apellido" : "Rangel",
  "edad" : 3,
  "cedula" : 1000000001,
  "telefono" : 6070000001,
  "direccion" : "condado campestre"
}
```