

# MongoDB

*Instalando e utilizando comandos*

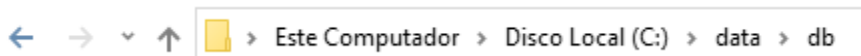
Bancos de dados não relacionais são aconselhados para sistemas que necessitam de muita velocidade (alto desempenho) no acesso à informação. Por outro lado, deixam a desejar na questão da integridade em decorrência da falta de relação.

Algumas das empresas que utilizam são [Amazon](#) e [Netflix](#).

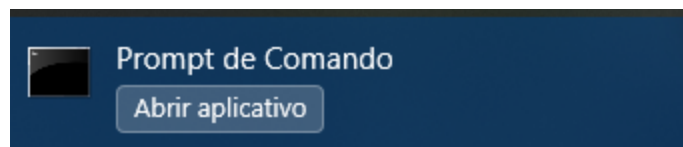
## Instalar o MongoDB (versão 5.0)

- **Acessa o site**  
<https://www.mongodb.com/try/download/community-edition>
- **Clica em “Select package”**
- **Seleciona a versão “5.0.22”**
- **Clica em Download**

## Criar pasta C:\data\db



## Abrir o CMD



## Acessar a pasta

- **Ir ao diretório inicial**  
`cd\`
- **Acessar “Arquivos de Programas”**  
`cd "Program Files"`
- **Acessar “MongoDB\Server”**  
`cd MongoDB\Server`
- **Lista as pastas dentro de “c:\Program Files\MongoDB\Server”**  
`dir`

- Entra na pasta 5.0 (ref. à versão 6.0 do MongoDB)

*cd 5.0*

- Acessa a pasta “bin”

*cd bin*

```
C:\WINDOWS\system32>cd\

C:\>cd "Program Files"

C:\Program Files>cd MongoDB\Server

C:\Program Files\MongoDB\Server>dir
O volume na unidade C não tem nome.
O Número de Série do Volume é 6420-F13E

Pasta de C:\Program Files\MongoDB\Server

31/10/2023  09:30    <DIR>          .
31/10/2023  09:30    <DIR>          ..
31/10/2023  09:30    <DIR>          5.0
31/10/2023  09:28    <DIR>          7.0
               0 arquivo(s)                0 bytes
               4 pasta(s) 168.735.866.880 bytes disponíveis

C:\Program Files\MongoDB\Server>cd 5.0

C:\Program Files\MongoDB\Server\5.0>cd bin
```

## Iniciar MongoDB

- Digita o comando para inicialização via linha de comando na pasta “data\db”

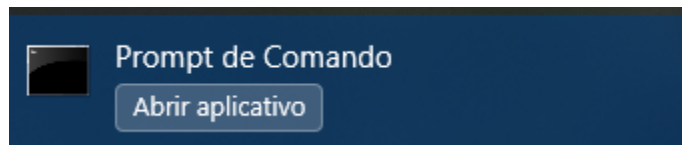
*mongod -dbpath c:\data\db*

**Aguarda a resposta, constando que iniciou**

```
Administrador: Prompt de Comando - mongod --dbpath C:\data\db
C:\Program Files\MongoDB\Server\5.0>cd bin
C:\Program Files\MongoDB\Server\5.0\bin>mongod --dbpath C:\data\db
{"t":{"$date":"2023-10-31T09:35:17.431-03:00"},"s":"I", "c":"NETWORK", "id":4915701, "ctx":"thread1","msg":"Initialize
d wire specification","attr":{"spec":{"incomingExternalClient":{"minWireVersion":0,"maxWireVersion":13},"incomingIntern
alClient":{"minWireVersion":0,"maxWireVersion":13},"outgoing":{"minWireVersion":0,"maxWireVersion":13},"isInternalClient"
:true}}}
{"t":{"$date":"2023-10-31T09:35:17.432-03:00"},"s":"I", "c":"CONTROL", "id":23285, "ctx":"thread1","msg":"Automatica
lly disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2023-10-31T09:35:17.434-03:00"},"s":"W", "c":"ASIO", "id":22601, "ctx":"thread1","msg":"No Transpo
rtLayer configured during NetworkInterface startup"}
{"t":{"$date":"2023-10-31T09:35:17.434-03:00"},"s":"I", "c":"NETWORK", "id":4648602, "ctx":"thread1","msg":"Implicit T
CP FastOpen in use."}
{"t":{"$date":"2023-10-31T09:35:17.438-03:00"},"s":"W", "c":"ASIO", "id":22601, "ctx":"thread1","msg":"No Transpo
rtLayer configured during NetworkInterface startup"}
{"t":{"$date":"2023-10-31T09:35:17.438-03:00"},"s":"I", "c":"REPL", "id":5123008, "ctx":"thread1","msg":"Successful
ly registered PrimaryOnlyService","attr":{"service":"TenantMigrationDonorService","ns":"config.tenantMigrationDonors"}}
{"t":{"$date":"2023-10-31T09:35:17.438-03:00"},"s":"I", "c":"REPL", "id":5123008, "ctx":"thread1","msg":"Successful
ly registered PrimaryOnlyService","attr":{"service":"TenantMigrationRecipientService","ns":"config.tenantMigrationRecipi
ents"}}
{"t":{"$date":"2023-10-31T09:35:17.439-03:00"},"s":"I", "c":"CONTROL", "id":5945603, "ctx":"thread1","msg":"Multi thre
ading initialized"}
{"t":{"$date":"2023-10-31T09:35:17.440-03:00"},"s":"I", "c":"CONTROL", "id":4615611, "ctx":"initandlisten","msg":"Mong
oDB starting","attr":{"pid":13296,"port":27017,"dbPath":"C:/data/db","architecture":"64-bit","host":"DPNote"}}
{"t":{"$date":"2023-10-31T09:35:17.440-03:00"},"s":"I", "c":"CONTROL", "id":23398, "ctx":"initandlisten","msg":"Targ
et operating system minimum version","attr":{"targetMinOS":"Windows 7/Windows Server 2008 R2"}}
{"t":{"$date":"2023-10-31T09:35:17.441-03:00"},"s":"I", "c":"CONTROL", "id":23403, "ctx":"initandlisten","msg":"Buil
d Info","attr":{"buildInfo":{"version":"5.0.22","gitVersion":"302c19437ae65b7d459360e18c5ac5086f494989","modules":[],"al
locator":"tcmalloc","environment":{"distmod":"windows","distarch":"x86_64","target_arch":"x86_64"}}}}}
{"t":{"$date":"2023-10-31T09:35:17.441-03:00"},"s":"I", "c":"CONTROL", "id":51765, "ctx":"initandlisten","msg":"Open
```

## Abre outro CMD

- **Abre outro CMD para iniciar a inserção de comandos**



## Acessa a pasta do MongoDB

- **Já no segundo CMD, vai para o diretório inicial**  
`cd\`
- **Agora, acessa “Arquivos de Programas”**  
`cd “Program Files”`
- **Entra na pasta do MongoDB**  
`cd MongoDB\Server\5.0\bin`

```
C:\Users\Diego>cd\  
C:\>cd "Program Files"  
C:\Program Files>cd MongoDB\Server\5.0\bin  
C:\Program Files\MongoDB\Server\5.0\bin>
```

## Acessa o Mongo

- **Digita o comando para acessar o banco de dados que foi inicializado no outro CMD**  
*mongo*

```
C:\Program Files\MongoDB\Server\5.0\bin>mongo  
MongoDB shell version v5.0.22  
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb  
Implicit session: session { "id" : UUID("6dd1141b-62da-4e26-b49a-62cbe407f055") }  
MongoDB server version: 5.0.22  
=====  
Warning: the "mongo" shell has been superseded by "mongosh",  
which delivers improved usability and compatibility. The "mongo" shell has been deprecated and will be removed in  
an upcoming release.  
For installation instructions, see  
https://docs.mongodb.com/mongodb-shell/install/  
=====  
---  
The server generated these startup warnings when booting:  
  2023-10-31T09:33:56.349-03:00: Access control is not enabled for the database. Read and write access to data and  
  configuration is unrestricted  
---  
>
```

## Verifica os bancos de dados existentes

- **Digita o comando para listar os bancos de dados existentes**  
*show databases*

```
> show databases  
admin      0.000GB  
config     0.000GB  
db_aula    0.000GB  
local      0.000GB  
>
```

## Acesso um dos bancos de dados

O único criado por usuário é o "db\_aula". O restante é criado pelo próprio MongoDB, automaticamente.

- **Acesso o “db\_aula”**

*use db\_aula*

```
> use db_aula
switched to db db_aula
>
```

Listo as tabelas do “db\_aula”

- **Já dentro da banco, listo as tabelas da mesma**

*show collections;*

```
> show collections;
dim_produto
fato_vendas
>
```

Deletar o banco de dados

- **Agora, vamos deletar o banco de dados “db\_aula”**

*db.dropDatabase();*

```
> db.dropDatabase();
{ "ok" : 1 }
>
```

Conferir se o banco de dados “db\_aula” foi excluído

- **Para conferir, utilizamos, novamente, o comando para listar os bancos de dados**

*show databases*

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

Crio novamente o banco de dados “db\_aula”

- **Através do comando abaixo, ele acessa o banco “db\_aula”. Se não existia, ele cria e já acessa, também**

*use db\_aula*

```
> use db_aula
switched to db db_aula
>
```

Já aparece na lista de banco de dados?

- **Ao utilizar o comando, ainda não constará, pois ainda não criei tabelas dentro dele**

*show databases*

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

Criando tabelas/registros

- **No MongoDB, para criar tabelas, basta inserir registros**

*db.fato\_vendas.insert({CodigoVenda:1, DataVenda:'2010-02-01 00:00:00', CodigoCliente:1, CodigoProduto:1, Quantidade: 4000, ValorUnitario:0.34, CodigoVendedor:2, CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});*

```
> db.fato_vendas.insert({CodigoVenda:1, DataVenda:'2010-02-01 00:00:00', CodigoCliente:1, CodigoProduto:1, Quantidade: 4000, ValorUnitario:0.34, CodigoVendedor:2, CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
WriteResult({ "nInserted" : 1 })
>
```

Verificar se agora aparece o banco de dados “db\_aula”

- **Para verificar, vamos, novamente, listar os bancos de dados existentes**

*show databases*

```
> show databases
admin    0.000GB
config   0.000GB
db_aula  0.000GB
local    0.000GB
>
```

## Listar tabelas do banco de dados “db\_aula”

- **Como já estamos dentro do “db\_aula”, o comando pode ser digitado diretamente**

*show collections*

```
> show collections
fato_vendas
>
```

## Inserir mais registros

- **Agora, vamos inserir mais registros**

```
db.fato_vendas.insert({CodigoVenda:2, DataVenda:'2010-02-01 00:00:00', CodigoCliente:1,
CodigoProduto:null, Quantidade: 4000, ValorUnitario:0.34, CodigoVendedor:2,
CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
```

```
db.fato_vendas.insert({CodigoVenda:3, DataVenda:'2010-02-01 00:00:00', CodigoCliente:5,
CodigoProduto:1, Quantidade: 4200, ValorUnitario:0.34, CodigoVendedor:4,
CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
```

```
db.fato_vendas.insert({CodigoVenda:4, DataVenda:'2010-02-01 00:00:00', CodigoCliente:3,
CodigoProduto:2, Quantidade: 250, ValorUnitario:7.00, CodigoVendedor:2,
CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
```

```
db.fato_vendas.insert({CodigoVenda:5, DataVenda:'2010-02-01 00:00:00', CodigoCliente:4,
CodigoProduto:1, Quantidade: 4500, ValorUnitario:0.34, CodigoVendedor:1,
CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
```

```
db.fato_vendas.insert({CodigoVenda:7, DataVenda:'2010-02-01 00:00:00', CodigoCliente:6,
CodigoProduto:1, Quantidade: 12500, ValorUnitario:0.34, CodigoVendedor:2,
CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
```

```
db.fato_vendas.insert({CodigoVenda:8, DataVenda:'2010-02-01 00:00:00', CodigoCliente:7,
CodigoProduto:1, Quantidade: 12000, ValorUnitario:0.34, CodigoVendedor:2,
CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
```

```
db.fato_vendas.insert({CodigoVenda:9, DataVenda:'2010-02-01 00:00:00', CodigoCliente:8,
CodigoProduto:1, Quantidade: 17500, ValorUnitario:0.34, CodigoVendedor:3,
CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
```

```
db.fato_vendas.insert({CodigoVenda:11, DataVenda:'2010-02-01 00:00:00', CodigoCliente:2,
CodigoProduto:1, Quantidade: 7000, ValorUnitario:0.34, CodigoVendedor:9,
CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
```

```
db.fato_vendas.insert({CodigoVenda:12, DataVenda:'2010-02-01 00:00:00', CodigoCliente:2,
CodigoProduto:1, Quantidade: 8000, ValorUnitario:0.34, CodigoVendedor:9,
CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
```

```
> db.fato_vendas.insert({CodigoVenda:2, DataVenda:'2010-02-01 00:00:00', CodigoCliente:1, CodigoProduto:null, Quantidade: 4000, ValorUnitario:0.34, CodigoVendedor:2, CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
WriteResult({ "nInserted" : 1 })
>
> db.fato_vendas.insert({CodigoVenda:3, DataVenda:'2010-02-01 00:00:00', CodigoCliente:5, CodigoProduto:1, Quantidade: 4200, ValorUnitario:0.34, CodigoVendedor:4, CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
WriteResult({ "nInserted" : 1 })
>
> db.fato_vendas.insert({CodigoVenda:4, DataVenda:'2010-02-01 00:00:00', CodigoCliente:3, CodigoProduto:2, Quantidade: 250, ValorUnitario:7.00, CodigoVendedor:2, CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
WriteResult({ "nInserted" : 1 })
>
> db.fato_vendas.insert({CodigoVenda:5, DataVenda:'2010-02-01 00:00:00', CodigoCliente:4, CodigoProduto:1, Quantidade: 4500, ValorUnitario:0.34, CodigoVendedor:1, CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
WriteResult({ "nInserted" : 1 })
>
> db.fato_vendas.insert({CodigoVenda:7, DataVenda:'2010-02-01 00:00:00', CodigoCliente:6, CodigoProduto:1, Quantidade: 12500, ValorUnitario:0.34, CodigoVendedor:2, CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
WriteResult({ "nInserted" : 1 })
>
> db.fato_vendas.insert({CodigoVenda:8, DataVenda:'2010-02-01 00:00:00', CodigoCliente:7, CodigoProduto:1, Quantidade: 12000, ValorUnitario:0.34, CodigoVendedor:2, CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
WriteResult({ "nInserted" : 1 })
>
> db.fato_vendas.insert({CodigoVenda:9, DataVenda:'2010-02-01 00:00:00', CodigoCliente:8, CodigoProduto:1, Quantidade: 17500, ValorUnitario:0.34, CodigoVendedor:3, CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
WriteResult({ "nInserted" : 1 })
>
> db.fato_vendas.insert({CodigoVenda:11, DataVenda:'2010-02-01 00:00:00', CodigoCliente:2, CodigoProduto:1, Quantidade: 7000, ValorUnitario:0.34, CodigoVendedor:9, CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
WriteResult({ "nInserted" : 1 })
>
> db.fato_vendas.insert({CodigoVenda:12, DataVenda:'2010-02-01 00:00:00', CodigoCliente:2, CodigoProduto:1, Quantidade: 8000, ValorUnitario:0.34, CodigoVendedor:9, CodigoCanalVendas:1, CodigoStatus:1, Deletado:0});
WriteResult({ "nInserted" : 1 })
```

## Como utilizar o Select

- Para usar o select, basta buscar o que está na tabela “fato\_vendas”

```
db.fato_vendas.find()
```

```
> db.fato_vendas.find()
{ "_id" : ObjectId("6541280825073ad54453882f"), "CodigoVenda" : 1, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 1, "CodigoProduto" : 1, "Quantidade" : 4000, "ValorUnitario" : 0.34, "CodigoVendedor" : 2, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0, "saldoProduto:5000, StatusProduto:'Ativo'"));
{ "_id" : ObjectId("65412a0625073ad544538830"), "CodigoVenda" : 2, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 1, "CodigoProduto" : null, "Quantidade" : 4000, "ValorUnitario" : 0.34, "CodigoVendedor" : 2, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0, "saldoProduto:2000, StatusProduto:'Ativo'"));
{ "_id" : ObjectId("65412a0625073ad544538831"), "CodigoVenda" : 3, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 5, "CodigoProduto" : 1, "Quantidade" : 4200, "ValorUnitario" : 0.34, "CodigoVendedor" : 4, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0, "saldoProduto:3000, StatusProduto:'MVP - Validação'"));
{ "_id" : ObjectId("65412a0625073ad544538832"), "CodigoVenda" : 4, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 3, "CodigoProduto" : 2, "Quantidade" : 250, "ValorUnitario" : 7.0, "CodigoVendedor" : 2, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538833"), "CodigoVenda" : 5, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 4, "CodigoProduto" : 1, "Quantidade" : 4500, "ValorUnitario" : 0.34, "CodigoVendedor" : 1, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538834"), "CodigoVenda" : 7, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 6, "CodigoProduto" : 1, "Quantidade" : 12500, "ValorUnitario" : 0.34, "CodigoVendedor" : 2, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538835"), "CodigoVenda" : 8, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 7, "CodigoProduto" : 1, "Quantidade" : 12000, "ValorUnitario" : 0.34, "CodigoVendedor" : 2, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538836"), "CodigoVenda" : 9, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 8, "CodigoProduto" : 1, "Quantidade" : 17500, "ValorUnitario" : 0.34, "CodigoVendedor" : 3, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538837"), "CodigoVenda" : 11, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 2, "CodigoProduto" : 1, "Quantidade" : 7000, "ValorUnitario" : 0.34, "CodigoVendedor" : 9, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538838"), "CodigoVenda" : 12, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 2, "CodigoProduto" : 1, "Quantidade" : 8000, "ValorUnitario" : 0.34, "CodigoVendedor" : 9, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
```

## Outra forma de utilizar o Select

- É possível usar o “select” de forma mais amigável

```
db.fato_vendas.find().pretty
```



[illegible]

## Filtrar as vendas de determinado cliente

- **Visualizar as vendas somente do cliente código "1"**

```
db.fato_vendas.find({CodigoCliente:1})
```

```
db.fato_vendas.find({CodigoCliente:1})
{ "_id" : ObjectId("6541280825073ad54453882f"), "CodigoVenda" : 1, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 1, "CodigoProduto" : 1, "Quantidade" : 4000, "ValorUnitario" : 0.34, "CodigoVendedor" : 2, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538830"), "CodigoVenda" : 2, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 1, "CodigoProduto" : null, "Quantidade" : 4000, "ValorUnitario" : 0.34, "CodigoVendedor" : 2, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
db.fato_vendas.find({CodigoCliente:1})
```

## Filtrar as vendas de quantidade “maior que”

- **Visualizar as vendas somente com quantidades acima de 4.000**

`db.fato_vendas.find({Quantidade: {$gt: 4000}})`

```
db.fato_vendas.find({Quantidade: {$gt: 4000}})
{ "_id" : ObjectId("65412a0625073ad544538831"), "CodigoVenda" : 3, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 5, "CodigoProduto" : 1, "Quantidade" : 4200, "ValorUnitario" : 0.34, "CodigoVendedor" : 4, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538833"), "CodigoVenda" : 5, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 4, "CodigoProduto" : 1, "Quantidade" : 4500, "ValorUnitario" : 0.34, "CodigoVendedor" : 1, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538834"), "CodigoVenda" : 7, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 6, "CodigoProduto" : 1, "Quantidade" : 12500, "ValorUnitario" : 0.34, "CodigoVendedor" : 2, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538835"), "CodigoVenda" : 8, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 7, "CodigoProduto" : 1, "Quantidade" : 12000, "ValorUnitario" : 0.34, "CodigoVendedor" : 2, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538836"), "CodigoVenda" : 9, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 8, "CodigoProduto" : 1, "Quantidade" : 17500, "ValorUnitario" : 0.34, "CodigoVendedor" : 3, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538837"), "CodigoVenda" : 11, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 2, "CodigoProduto" : 1, "Quantidade" : 7000, "ValorUnitario" : 0.34, "CodigoVendedor" : 9, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
{ "_id" : ObjectId("65412a0625073ad544538838"), "CodigoVenda" : 12, "DataVenda" : "2010-02-01 00:00:00", "CodigoCliente" : 2, "CodigoProduto" : 1, "Quantidade" : 8000, "ValorUnitario" : 0.34, "CodigoVendedor" : 9, "CodigoCanalVendas" : 1, "CodigoStatus" : 1, "Deletado" : 0 }
```

## Outra forma de filtrar as vendas de quantidade “maior que”

- **Visualizar as vendas somente com quantidades acima de 4.000, só que de outro formato**

`db.fato_vendas.find({Quantidade: {$gt: 4000}}).pretty()`

```

db.Fato_vendas.Find({Quantidade: {$gt: 4000}}).pretty()
{
  "_id" : ObjectId("65412a0625073ad544538831"),
  "CodigoVenda" : 3,
  "DataVenda" : "2010-02-01 00:00:00",
  "CodigoCliente" : 5,
  "CodigoProduto" : 1,
  "Quantidade" : 4200,
  "ValorUnitario" : 0.34,
  "CodigoVendedor" : 4,
  "CodigoCanalVendas" : 1,
  "CodigoStatus" : 1,
  "Deletado" : 0
}

{
  "_id" : ObjectId("65412a0625073ad544538833"),
  "CodigoVenda" : 5,
  "DataVenda" : "2010-02-01 00:00:00",
  "CodigoCliente" : 4,
  "CodigoProduto" : 1,
  "Quantidade" : 4500,
  "ValorUnitario" : 0.34,
  "CodigoVendedor" : 1,
  "CodigoCanalVendas" : 1,
  "CodigoStatus" : 1,
  "Deletado" : 0
}

{
  "_id" : ObjectId("65412a0625073ad544538834"),
  "CodigoVenda" : 7,
  "DataVenda" : "2010-02-01 00:00:00",
  "CodigoCliente" : 6,
  "CodigoProduto" : 1,
  "Quantidade" : 12500,
  "ValorUnitario" : 0.34,
  "CodigoVendedor" : 2,
  "CodigoCanalVendas" : 1,
  "CodigoStatus" : 1,
  "Deletado" : 0
}

{
  "_id" : ObjectId("65412a0625073ad544538835"),
  "CodigoVenda" : 8,
  "DataVenda" : "2010-02-01 00:00:00",
  "CodigoCliente" : 7,
  "CodigoProduto" : 1,
  "Quantidade" : 12000,
  "ValorUnitario" : 0.34,
  "CodigoVendedor" : 2,
  "CodigoCanalVendas" : 1,
  "CodigoStatus" : 1,
  "Deletado" : 0
}

{
  "_id" : ObjectId("65412a0625073ad544538836"),
  "CodigoVenda" : 9,
  "DataVenda" : "2010-02-01 00:00:00",
  "CodigoCliente" : 8,
  "CodigoProduto" : 1,
  "Quantidade" : 17500,
  "ValorUnitario" : 0.34,
  "CodigoVendedor" : 3,
  "CodigoCanalVendas" : 1,
  "CodigoStatus" : 1,
  "Deletado" : 0
}

{
  "_id" : ObjectId("65412a0625073ad544538837"),
  "CodigoVenda" : 11,
  "DataVenda" : "2010-02-01 00:00:00",
  "CodigoCliente" : 2,
  "CodigoProduto" : 1,
  "Quantidade" : 7000,
  "ValorUnitario" : 0.34,
  "CodigoVendedor" : 9,
  "CodigoCanalVendas" : 1,
  "CodigoStatus" : 1,
  "Deletado" : 0
}

{
  "_id" : ObjectId("65412a0625073ad544538838"),
  "CodigoVenda" : 12,
  "DataVenda" : "2010-02-01 00:00:00",
  "CodigoCliente" : 2,
  "CodigoProduto" : 1,
  "Quantidade" : 8000,
  "ValorUnitario" : 0.34,
  "CodigoVendedor" : 9,
  "CodigoCanalVendas" : 1,
  "CodigoStatus" : 1,
  "Deletado" : 0
}
}

```

Criar nova tabela

- Criar tabela “dim\_produto”

```
db.dim_produto.insert({CodigoProduto:1, NomeProduto:'Produto A', TipoProduto:'A',  
UnidadeProduto:'KG', SaldoProduto:20000, StatusProduto:'Ativo'});
```

```
> db.dim_produto.insert({CodigoProduto:1, NomeProduto:'Produto A', TipoProduto:'A', UnidadeProduto:'KG', SaldoProduto:20000, StatusProduto:'Ativo'});  
WriteResult({ "nInserted" : 1 })
```

## Inserir mais registros

- **Inserir mais 5 registros dentro da tabela “dim\_produto”**

```
db.dim_produto.insert({CodigoProduto:2, NomeProduto:'Produto C', TipoProduto:'A',  
UnidadeProduto:'KG', SaldoProduto:5000, StatusProduto:'Ativo'});
```

```
db.dim_produto.insert({CodigoProduto:3, NomeProduto:'Produto E', TipoProduto:'B',  
UnidadeProduto:'KG', SaldoProduto:2000, StatusProduto:'Ativo'});
```

```
db.dim_produto.insert({CodigoProduto:5, NomeProduto:'Produto CH', TipoProduto:'A',  
UnidadeProduto:'KG', SaldoProduto:2000, StatusProduto:'Ativo'});
```

```
db.dim_produto.insert({CodigoProduto:4, NomeProduto:'Produto SL', TipoProduto:'A',  
UnidadeProduto:'KG', SaldoProduto:30000, StatusProduto:'Ativo'});
```

```
db.dim_produto.insert({CodigoProduto:6, NomeProduto:'Produto TN', TipoProduto:'C',  
UnidadeProduto:'KG', SaldoProduto:30000, StatusProduto:'MVP - Validação'});
```

```
> db.dim_produto.insert({CodigoProduto:1, NomeProduto:'Produto A', TipoProduto:'A', UnidadeProduto:'KG', SaldoProduto:20000, StatusProduto:'Ativo'});  
WriteResult({ "nInserted" : 1 })  
> db.dim_produto.insert({CodigoProduto:2, NomeProduto:'Produto C', TipoProduto:'A', UnidadeProduto:'KG', SaldoProduto:5000, StatusProduto:'Ativo'});  
WriteResult({ "nInserted" : 1 })  
>  
> db.dim_produto.insert({CodigoProduto:3, NomeProduto:'Produto E', TipoProduto:'B', UnidadeProduto:'KG', SaldoProduto:2000, StatusProduto:'Ativo'});  
WriteResult({ "nInserted" : 1 })  
>  
> db.dim_produto.insert({CodigoProduto:5, NomeProduto:'Produto CH', TipoProduto:'A', UnidadeProduto:'KG', SaldoProduto:2000, StatusProduto:'Ativo'});  
WriteResult({ "nInserted" : 1 })  
>  
> db.dim_produto.insert({CodigoProduto:4, NomeProduto:'Produto SL', TipoProduto:'A', UnidadeProduto:'KG', SaldoProduto:30000, StatusProduto:'Ativo'});  
WriteResult({ "nInserted" : 1 })  
>  
> db.dim_produto.insert({CodigoProduto:6, NomeProduto:'Produto TN', TipoProduto:'C', UnidadeProduto:'KG', SaldoProduto:30000, StatusProduto:'MVP - Validação'});  
WriteResult({ "nInserted" : 1 })  
>
```

## Consultar tabela

- **Vamos verificar se os registros estão constando na tabela “dim\_produto”**

```
db.dim_produto.find().pretty()
```

```

> db.dim_produto.find().pretty()
{
  "_id" : ObjectId("654130e225073ad544538839"),
  "CodigoProduto" : 1,
  "NomeProduto" : "Produto A",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 20000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883a"),
  "CodigoProduto" : 2,
  "NomeProduto" : "Produto C",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 5000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883b"),
  "CodigoProduto" : 3,
  "NomeProduto" : "Produto E",
  "TipoProduto" : "B",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 2000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883c"),
  "CodigoProduto" : 5,
  "NomeProduto" : "Produto CH",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 2000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883d"),
  "CodigoProduto" : 4,
  "NomeProduto" : "Produto SL",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 30000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883e"),
  "CodigoProduto" : 6,
  "NomeProduto" : "Produto TN",
  "TipoProduto" : "C",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 30000,
  "StatusProduto" : "MVP - Validação"
}
>

```

Trazer os primeiros registros

- **Puxar somente os 03 (três) primeiros registros da tabela “dim\_produto”**

*db.dim\_produto.find().limit(3)*

```
> db.dim_produto.find().limit(3)
{ "_id" : ObjectId("654130e225073ad544538839"), "CodigoProduto" : 1, "NomeProduto" : "Produto A", "TipoProduto" : "A", "UnidadeProduto" : "KG", "SaldoProduto" : 20000, "StatusProduto" : "Ativo" }
{ "_id" : ObjectId("6541318525073ad54453883a"), "CodigoProduto" : 2, "NomeProduto" : "Produto C", "TipoProduto" : "A", "UnidadeProduto" : "KG", "SaldoProduto" : 5000, "StatusProduto" : "Ativo" }
{ "_id" : ObjectId("6541318525073ad54453883b"), "CodigoProduto" : 3, "NomeProduto" : "Produto E", "TipoProduto" : "B", "UnidadeProduto" : "KG", "SaldoProduto" : 2000, "StatusProduto" : "Ativo" }
```

Outra forma de trazer os primeiros registros

- **Puxar os três primeiros registros, porém, em um formato mais amigável**

`db.dim_produto.find().limit(3).pretty()`

```
> db.dim_produto.find().limit(3).pretty()
{
  "_id" : ObjectId("654130e225073ad544538839"),
  "CodigoProduto" : 1,
  "NomeProduto" : "Produto A",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 20000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883a"),
  "CodigoProduto" : 2,
  "NomeProduto" : "Produto C",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 5000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883b"),
  "CodigoProduto" : 3,
  "NomeProduto" : "Produto E",
  "TipoProduto" : "B",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 2000,
  "StatusProduto" : "Ativo"
}
>
```

Listar produtos com base no Saldo

- **Listar apenas produtos que possuem um saldo entre 2000 e 5000**

`db.dim_produto.find( { "SaldoProduto" : { "$gte" : 2000 , "$lte" : 5000 } }).pretty();`

```
> db.dim_produto.find( { "SaldoProduto" : { "$gte" : 2000 , "$lte" : 5000}}).pretty();
{
  "_id" : ObjectId("6541318525073ad54453883a"),
  "CodigoProduto" : 2,
  "NomeProduto" : "Produto C",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 5000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883b"),
  "CodigoProduto" : 3,
  "NomeProduto" : "Produto E",
  "TipoProduto" : "B",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 2000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883c"),
  "CodigoProduto" : 5,
  "NomeProduto" : "Produto CH",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 2000,
  "StatusProduto" : "Ativo"
}
```

## Adicionar coluna à tabela já criada

- **Em bancos não relacionais é possível adicionar colunas usando tão somente o comando de inserção**

```
db.dim_produto.insert({CodigoProduto:9, NomeProduto:null, TipoProduto:'A',
UnidadeProduto:'KG', SaldoProduto:20000, StatusProduto:'Inativo', info:{motivo:"Sem
movimentação", data:'2022-05-29 13:45:00'}});
```

```
> db.dim_produto.insert({CodigoProduto:9, NomeProduto:null, TipoProduto:'A', UnidadeProduto:'KG', SaldoProduto:20000, StatusProduto:'Inativo', info:{motivo:"Sem movimentação", data:'2022-05-29 13:45:00'}});
WriteResult({ "nInserted" : 1 })
>
```

## Verificar se a coluna já foi adicionada à tabela

- **Para verificar se a coluna “info” foi criada, basta dar um select na tabela “dim\_produto”**

```
db.dim_produto.find().pretty()
```

```

> db.dim_produto.find().pretty()
{
  "_id" : ObjectId("654130e225073ad544538839"),
  "CodigoProduto" : 1,
  "NomeProduto" : "Produto A",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 20000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883a"),
  "CodigoProduto" : 2,
  "NomeProduto" : "Produto C",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 5000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883b"),
  "CodigoProduto" : 3,
  "NomeProduto" : "Produto E",
  "TipoProduto" : "B",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 2000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883c"),
  "CodigoProduto" : 5,
  "NomeProduto" : "Produto CH",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 2000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883d"),
  "CodigoProduto" : 4,
  "NomeProduto" : "Produto SL",
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 30000,
  "StatusProduto" : "Ativo"
}
{
  "_id" : ObjectId("6541318525073ad54453883e"),
  "CodigoProduto" : 6,
  "NomeProduto" : "Produto TN",
  "TipoProduto" : "C",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 30000,
  "StatusProduto" : "MVP - Validação"
}
{
  "_id" : ObjectId("654141ab25073ad54453883f"),
  "CodigoProduto" : 9,
  "NomeProduto" : null,
  "TipoProduto" : "A",
  "UnidadeProduto" : "KG",
  "SaldoProduto" : 20000,
  "StatusProduto" : "Inativo",
  "info" : {
    "motivo" : "Sem movimentação",
    "data" : "2022-05-29 13:45:00"
  }
}
>

```



## Consultar o “relacionamento” entre as tabelas

- Não é bem relacionamento, já que o MongoDB é não relacional, mas o comando “lookup” mostra algum tipo de ligação entre as tabelas

*db.fato\_vendas.aggregate( [ { \$lookup: { from: 'dim\_produto', localField: 'CodigoProduto', foreignField: 'CodigoProduto', as: 'detalhe\_produto' } } ] ).pretty();*

```
> db.fato_vendas.aggregate( [ { $lookup: { from: 'dim_produto', localField: 'CodigoProduto', foreignField: 'CodigoProduto', as: 'detalhe_produto' } } ] ).pretty();
{
  "_id" : ObjectId("6541280825073ad54453882f"),
  "CodigoVenda" : 1,
  "DataVenda" : "2010-02-01 00:00:00",
  "CodigoCliente" : 1,
  "CodigoProduto" : 1,
  "Quantidade" : 4000,
  "ValorUnitario" : 0.34,
  "CodigoVendedor" : 2,
  "CodigoCanalVendas" : 1,
  "CodigoStatus" : 1,
  "Deletado" : 0,
  "detalhe_produto" : [
    {
      "_id" : ObjectId("654130e225073ad544538839"),
      "CodigoProduto" : 1,
      "NomeProduto" : "Produto A",
      "TipoProduto" : "A",
      "UnidadeProduto" : "KG",
      "SaldoProduto" : 20000,
      "StatusProduto" : "Ativo"
    }
  ]
}
```

