



Desenvolvimento Web— Teoria

Angelina V S Melaré | FACENS
angelina.melare@facens.br



MONGO DB

ANGELINA V S MELARÉ



Instalação do Mongo

MongoDB é o sistema gerenciador de banco de dados (SGBD). Para sua importação e exportação é usado o MongoDB Tools.

Será usada uma versão flexível, a Community. Com o MongoDB Atlas acesso a funcionalidades avançadas, como dimensionamento automático, instâncias sem servidor (em visualização), pesquisa de texto completo e ainda pode fazer a implantação na AWS, Google Cloud e/ou Azure, sem necessidade de downloads. Ainda se pode ter disponível o banco de dados como um serviço com um cluster de 512 MB gratuito.

As ferramentas de banco de dados do MongoDB são uma coleção de utilitários de linha de comando para trabalhar com uma implantação do MongoDB.

<https://www.mongodb.com/pt-br>

<https://www.mongodb.com/try/download/community>





Atlas

MongoDB as a service



On-premises

MongoDB locally



Tools

Boost productivity



Mobile & Edge

Realm Datastore

MongoDB Community Server

The Community version of our distributed database offers a flexible document data model along with support for ad-hoc queries, secondary indexing, and real-time aggregations to provide powerful ways to access and analyze your data.

The database is also offered as a fully-managed service with [MongoDB Atlas](#). Get access to advanced functionality such as auto-scaling, serverless instances (in preview), full-text search,

Available Downloads

Version

5.0.8 (current)

Platform

Windows

Package

msi



Download

Copy Link

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

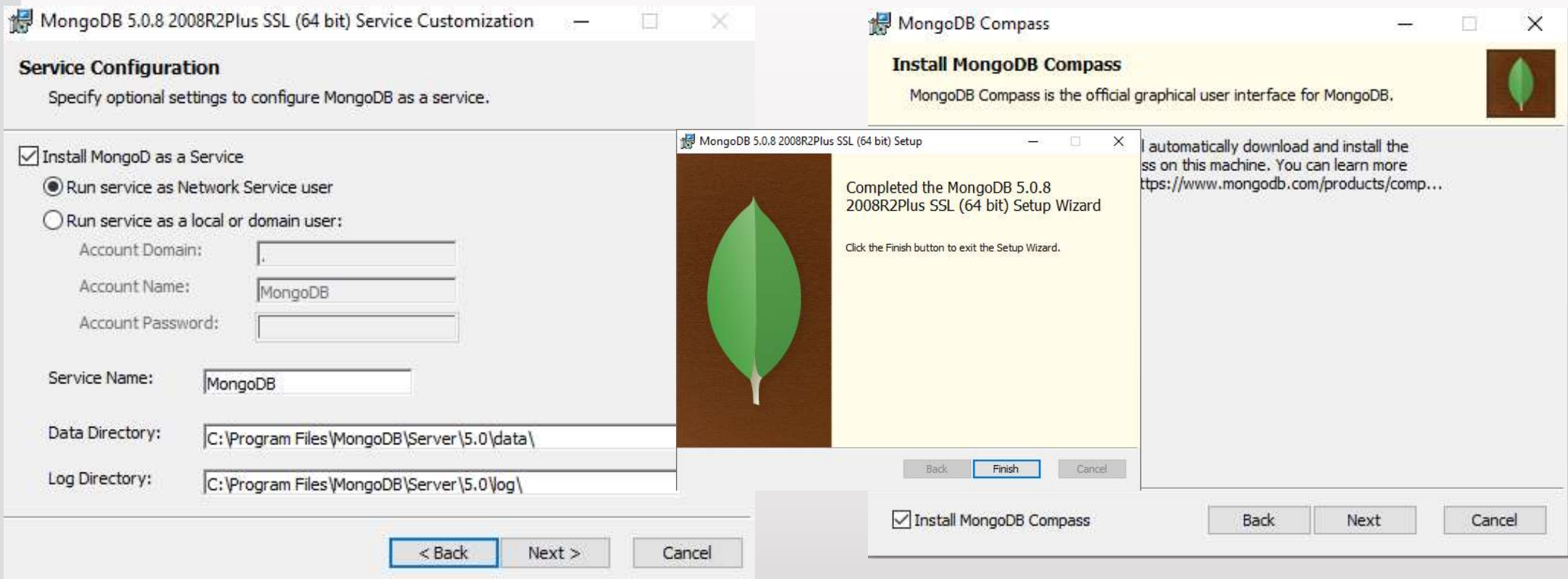
zip



Download

Copy Link

Instalando a versão completa do Mongo



Versão Compass é a versão gráfica do MongoDB

New Connection +

☆ Favorites

🕒 Recents

New Connection

Connect to a MongoDB deployment

URI ⓘ

mongodb://

> Advanced

Privacy Settings

To enhance the user experience, Compass can integrate with 3rd party services, which requires external network requests. Please choose from the settings below:

☒ Enable Automatic Updates

Allow Compass to periodically check for updates

☒ Enable Geographic Visualizations

Allow Compass to make requests to a 3rd party service

☒ Enable Crash Reports

Allow Compass to send crash reports to MongoDB

☒ Enable Usage Statistics

Allow Compass to send anonymous usage data to MongoDB

☒ Give Product Feedback

Enables a tool that our Product team can use to collect feedback from Compass.

With any of these options, none of your personal information or stored data will be submitted.

Learn more: [MongoDB Privacy Policy](#)

New Connection

Connect to a MongoDB deployment

URI ⓘ

Edit Connection String ☒

mongodb://localhost:27017

> Advanced Connection Options



FAVORITE

Connect

Start Using Compass

MongoDB Compass - localhost:27017

Connect View Help

localhost:27017

3 DBS 3 COLLECTIONS

☆ FAVORITE

HOST
localhost:27017

CLUSTER
Standalone

EDITION
MongoDB 5.0.8 Community

{ } My Queries

☰ Databases

🔍 Filter your data

▸ admin

▸ config

▸ local

My Queries

Databases

Performance

Create database

View



admin

Storage size:
20.48 kB

Collections:
1

Indexes:
1

config

Storage size:
4.10 kB

Collections:
1

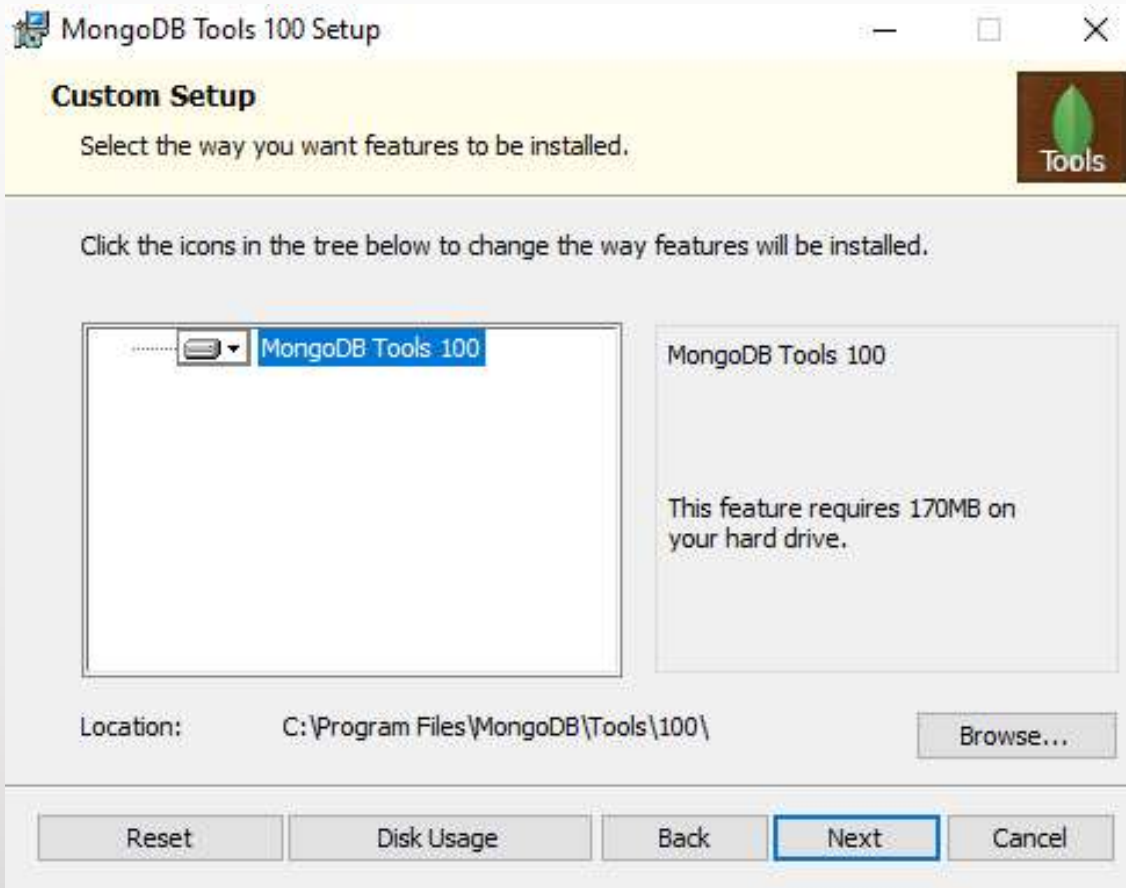
Indexes:
2

local

Storage size:
20.48 kB

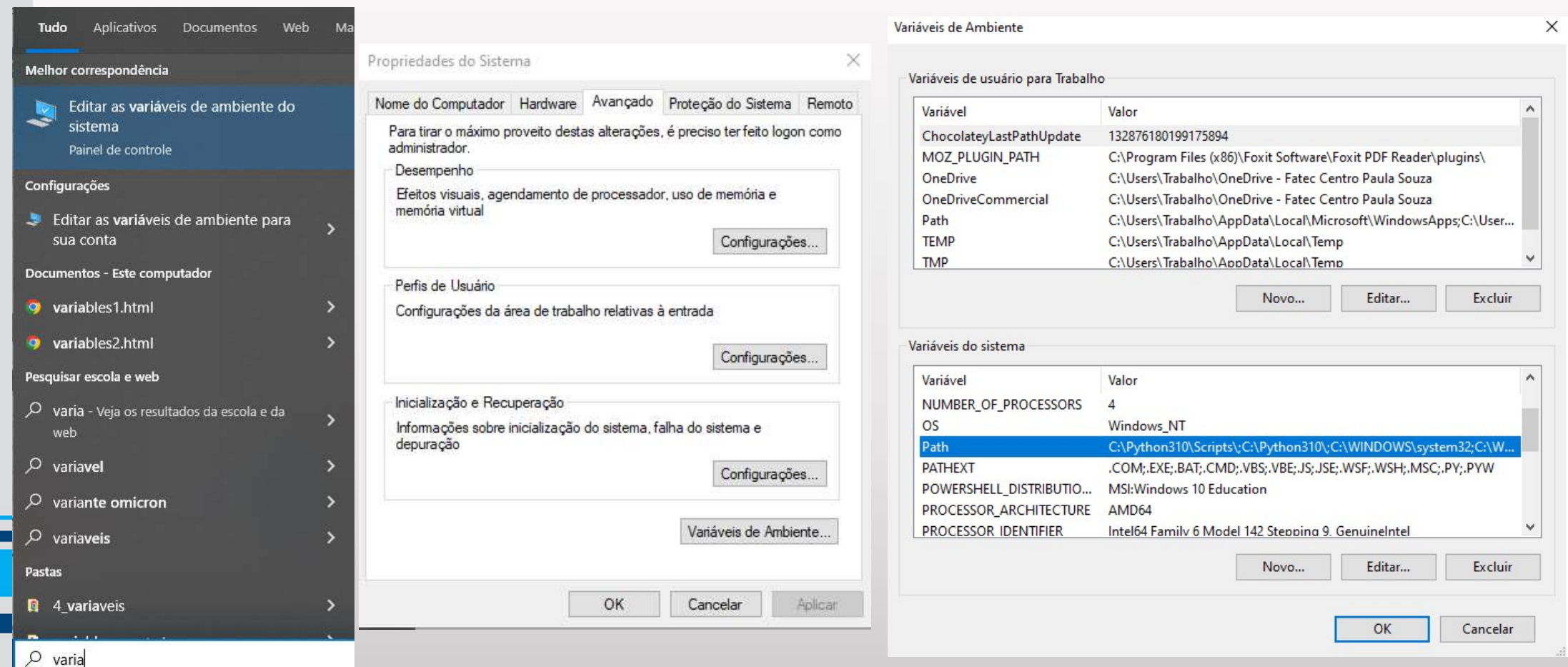
Collections:
1

Indexes:
1



Configurando as variáveis de ambiente

Para reconhecer o mongo e/ou outros programas verifique se elas tem suas variáveis ambientes



Editando as variáveis ambiente

Editar a variável de ambiente

C:\Python310\Scripts\
C:\Python310\
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
%SYSTEMROOT%\System32\OpenSSH\
C:\Program Files\dotnet\
C:\Program Files\Microsoft SQL Server\130\Tools\Binn\
C:\Program Files\Microsoft SQL Server\Client SDK\ODBC\170\Tools\Bi...
C:\Program Files (x86)\dotnet\
C:\Program Files\nodejs\
C:\ProgramData\chocolatey\bin
C:\Program Files\Git\cmd
C:\Program Files\PowerShell\7\
C:\Program Files\MongoDB\Server\5.0\bin
C:\Program Files\MongoDB\Tools\100\bin

Novo

Editar

Procurar...

Excluir

Mover para Cima

Mover para baixo

Editar texto...

OK

Cancelar

Dentro do path serão acrescidos os endereços do MongoDB

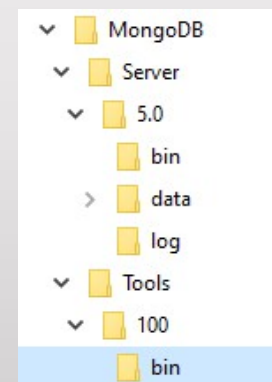
Para isso primeiro pegue a pasta onde se encontra a pasta BIN instalada do Server do MongoDB

C:\Program Files\MongoDB\Server\5.0\bin

Para armazenar esse endereço clique no botão “Novo” e cole o endereço copiado

Faça isso para a MongoDB Tools

C:\Program Files\MongoDB\Tools\100\bin



Se ocorrer erro nos testes edite o path da variável do usuário

Variáveis de usuário para Trabalho

Variável	Valor
ChocolateyLastPathUpdate	132876180199175894
MOZ_PLUGIN_PATH	C:\Program Files (x86)\Foxit Software\Foxit PDF Reader\Plugins
OneDrive	C:\Users\Trabalho\OneDrive - Fatec Centro Paula Souza
OneDriveCommercial	C:\Users\Trabalho\OneDrive - Fatec Centro Paula Souza
Path	C:\Users\Trabalho\AppData\Local\Microsoft\Windows\CurrentVersion\Path

Testando

C:\> Prompt de Comando - mongo

```
Microsoft Windows [versão 10.0.19044.1645]  
(c) Microsoft Corporation. Todos os direitos reservados.  
  
C:\Users\Trabalho>mongo  
MongoDB shell version v5.0.8  
connecting to: mongodb://127.0.0.1:27017/?readPreference=primary&w=1  
Implicit session: session { "id" : UUID("29...") }  
MongoDB server version: 5.0.8
```

No CMD digite
Mongo e veja
se ele criou
uma conexão
com servidor

Testando serviço

```
C:\Users\Trabalho>mongod
{"t":{"$date":"2022-05-01T22:04:48.097-03:00"},"s":4915701, "ctx":"-", "msg":"Initialized wire specification", "info":{"externalClient":{"minWireVersion":0,"maxWireVersion":13},"internalClient":{"minWireVersion":0,"maxWireVersion":13},"isInternalClient":true}}
```

Shell MongoDB

Pelo CMD ou outro console é possível executar o shell MongoDB, onde é possível interagir com o MongoDB por linha de comando.

Para isso, o servidor MongoDB deve estar executando.

MongoDB - Documentos

O MongoDB armazena registros de dados como **documentos BSON**.
O formato BSON é uma representação binária de documentos JSON.
Os documentos são compostos por campo e valores:

```
db.users.insertOne(  ← collection
{
  name: "sue",        ← field: value
  age: 26,             ← field: value
  status: "pending"   ← field: value
}
) }
```

MongoDB - Documentos

O valor de um campo pode ser qualquer um dos tipos de dados BSON, incluindo outros documentos, matrizes e matrizes de documentos.

O BSON tem mais tipos de dados do que JSON.

(<https://www.mongodb.com/docs/manual/reference/bson-types/>)

```
var mydoc = {  
  _id: ObjectId("5099803df3f4948bd2f98391"),  
  name: { first: "Alan", last: "Turing" },  
  birth: new Date('Jun 23, 1912'),  
  death: new Date('Jun 07, 1954'),  
  contribs: [ "Turing machine", "Turing test", "Turingery" ],  
  views : NumberLong(1250000)  
}
```

<https://www.mongodb.com/docs/manual/reference/bson-types/>

MongoDB - Documentos

```
{
  nome: "Angelina",
  idade: 48,
  email: "angelinamelare@gmail.com",
  dataCad: new Date(),
  endereco: {
    rua: "Rua Sorocaba, 220",
    cidade: "Sorocaba"
  },
  passatempo: ['artesanato', 'quebra-cabeça',
'jardinagem'],
  inadimplente: true
}
```

Criando o Banco de Dados

use clientes --- ele só efetiva a criação após a inserção de algum dado

```
> use clientes  
switched to db clientes
```

******não usa create database ...

show dbs – mostra todos os bancos de dados e você vê que o banco ainda não existe

Criando Documento de Dados - Collection

O MongoDB cria base de dados a partir de coleção de documentos (Collections)

Para inserir um dado não precisa de uma tabela e sim uma **Collection**. No Mongo ao inserir o dado já se cria a Collection (não precisa ser ao contrário igual num relacional).

```
> show collections
clientesCad
fornecedoresCad
>
```

Inserindo dados e criando a collection

Para se referir ao banco atual se coloca **db.** e depois os comandos que se quer aplicar.

db.pessoas.insertOne() – para inserir dados

O insertOne precisa de um documento, como um objeto JS, onde se insere os parâmetros, chave e valor

{nome: "Julia", telefone:"99844-2122", curso:"EC", idade:49, frequenta: true}

```
> db.clientesCad.insertOne({nome:"Angelina", telefone:"9677-2323", idade: 49,
curso: "EC", inadimplente: false})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("626f33cfe5d4d0616e45b8d0")
}
```


Inserindo dados e criando a collection

Ele irá criar um Id com uma string com um hash única.

Para ver pode colocar **show dbs** ou **show collections**

```
> show collections
clientesCad
```

```
> db.fornecedoresCad.insertOne({nome:"FA", telefone:"9677-2323"})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("626f34e7e5d4d0616e45b8d1")
}
> show collections
clientesCad
fornecedoresCad
```

Insertão de dados

insertOne() – Inserir apenas registros

InsertMany()- Inserir vários registros. Nessa situação usa o padrão de definição de um array de objetos [{},{}]

```
> db.fornecedoresCad.insertMany([{nome:"IT", telefone:"9677-2323"}, {nome: "SH", telefone:"5656-2323"}])
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("626f3609e5d4d0616e45b8d2"),
    ObjectId("626f3609e5d4d0616e45b8d3")
  ]
}
```

Consulta

O find é usado para consultar os dados cadastrados

db.clientesCad.find()

```
> db.clientesCad.find()
{ "_id" : ObjectId("626f33cfe5d4d0616e45b8d0"), "nome" : "Angelina", "telefone" : "9677-2323", "idade" : 49, "curso" : "EC", "inadimplente" : false }
> db.fornecedoresCad.find()
{ "_id" : ObjectId("626f34e7e5d4d0616e45b8d1"), "nome" : "FA", "telefone" : "9677-2323" }
{ "_id" : ObjectId("626f3609e5d4d0616e45b8d2"), "nome" : "IT", "telefone" : "9677-2323" }
{ "_id" : ObjectId("626f3609e5d4d0616e45b8d3"), "nome" : "SH", "telefone" : "5656-2323" }
>
```

Consulta

O pretty é usado para melhorar a exibição da consulta
db.fornecedoresCad.find().pretty()

```
> db.fornecedoresCad.find().pretty()
{
  "_id" : ObjectId("626f34e7e5d4d0616e45b8d1"),
  "nome" : "FA",
  "telefone" : "9677-2323"
}
{
  "_id" : ObjectId("626f3609e5d4d0616e45b8d2"),
  "nome" : "IT",
  "telefone" : "9677-2323"
}
{
  "_id" : ObjectId("626f3609e5d4d0616e45b8d3"),
  "nome" : "SH",
  "telefone" : "5656-2323"
}
```


Consulta com filtro

Pode-se definir filtros na consulta, como se fosse um Where.

db.clientesCad.find({inadimplente:true})

Para contar coloque o método count no final da pesquisa.

db.clientesCad.find({inadimplente:false}).count()

OU ainda o total de cadastrados, se não definir filtro

db.fornecedoresCad.find().count()

```
> db.clientesCad.find({inadimplente:true})
> db.clientesCad.find({inadimplente:false})
{ "_id" : ObjectId("626f33cfe5d4d0616e45b8d0"), "nome" : "9677-2323", "idade" : 49, "curso" : "EC", "inadimplente" : false }
> db.clientesCad.find({inadimplente:true}).count()
0
> db.clientesCad.find({inadimplente:false}).count()
1
```

Consulta

O findOne trás o primeiro que encontrar, mas apenas um retorno.

```
> db.clientesCad.findOne({idade: 18})
{
  "_id" : ObjectId("626f39a4e5d4d0616e45b8d4"),
  "nome" : "Bianca",
  "telefone" : "9677-2323",
  "idade" : 18,
  "curso" : "EC",
  "inadimplente" : false
}
> db.clientesCad.findOne({inadimplente: false})
{
  "_id" : ObjectId("626f33cfe5d4d0616e45b8d0"),
  "nome" : "Angelina",
  "telefone" : "9677-2323",
  "idade" : 49,
  "curso" : "EC",
  "inadimplente" : false
}
```

Organização

O método `sort` classifica os documentos em ordem, ascendente ou descendente.

Pode ser usada uma chave de classificação, onde esta é o campo que deseja e o 1 represente ordem ascendente, para descendente é o -1.

```
collectionName.find().sort({key:1})
```

```
> db.fornecedoresCad.find().sort({nome:1})
{ "_id" : ObjectId("626f34e7e5d4d0616e45b8d1"), "nome" : "FA", "telefone" : "9677-2323" }
{ "_id" : ObjectId("626f3609e5d4d0616e45b8d2"), "nome" : "IT", "telefone" : "9677-2323" }
{ "_id" : ObjectId("626f3609e5d4d0616e45b8d3"), "nome" : "SH", "telefone" : "5656-2323" }
> db.fornecedoresCad.find().sort({nome:-1})
{ "_id" : ObjectId("626f3609e5d4d0616e45b8d3"), "nome" : "SH", "telefone" : "5656-2323" }
{ "_id" : ObjectId("626f3609e5d4d0616e45b8d2"), "nome" : "IT", "telefone" : "9677-2323" }
{ "_id" : ObjectId("626f34e7e5d4d0616e45b8d1"), "nome" : "FA", "telefone" : "9677-2323" }
>
```

Atualização de dados

Para atualizar os dados é usado o **updateOne()** ou **updateMany()**.

* Primeiro defina um documento que define quem procura achar e depois outro documento que informa qual a mudança por meio do \$set.

```
db.nome.updateOne( {o que procura}, { $set: {qual a mudança} } )
```

```
> db.clientesCad.updateOne({nome:"Juca"},{$set: {"inadimplente":false} })  
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
```

Atualização de vários

Veja que não tem um valor de pesquisa por isso o primeiro documento é { }

db.clientesCad.updateMany({ },{\$set: {"inadimplente":true} })

```
> db.clientesCad.updateMany({ },{$set: {"inadimplente":true} })
{ "acknowledged" : true, "matchedCount" : 3, "modifiedCount" : 3 }
> db.clientesCad.find().pretty()
{
  "_id" : ObjectId("626f33cfe5d4d0616e45b8d0"),
  "nome" : "Angelina",
  "telefone" : "9677-2323",
  "idade" : 49,
  "curso" : "EC",
  "inadimplente" : true
}
{
  "_id" : ObjectId("626f39a4e5d4d0616e45b8d4"),
  "nome" : "Bianca",
  "telefone" : "9677-2323",
  "idade" : 18,
  "curso" : "EC",
  "inadimplente" : true
}
{
  "_id" : ObjectId("626f39c3e5d4d0616e45b8d5"),
  "nome" : "Juca",
  "telefone" : "9677-2323",
  "idade" : 18,
  "curso" : "EC",
  "inadimplente" : true
}
>
```


Atualização da estrutura

O `updateMany()` também pode ser usado para alterar a estrutura da collection, inserindo um novo campo.

```
> db.clientesCad.updateMany({ },{$set: {uf:"SP" } })
{ "acknowledged" : true, "matchedCount" : 3, "modifiedCount" : 3 }
> db.clientesCad.updateMany({ },{$set: {uf:"" } })
{ "acknowledged" : true, "matchedCount" : 3, "modifiedCount" : 3 }
> db.clientesCad.updateMany({ },{$set: {cidade:"" } })
{ "acknowledged" : true, "matchedCount" : 3, "modifiedCount" : 3 }
> db.clientesCad.find().pretty()
{
  "_id" : ObjectId("626f33cfe5d4d0616e45b8d0"),
  "nome" : "Angelina",
  "telefone" : "9677-2323",
  "idade" : 49,
  "curso" : "EC",
  "inadimplente" : true,
  "uf" : "",
  "cidade" : ""
}
```


Estrutura distintas

O MongoDB pode ter registros com campos diferentes

```
> db.clientesCad.updateOne({nome:"Angelina"},{ $set: {profissao:"professora" }} )
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.clientesCad.find().pretty()
{
  "_id" : ObjectId("626f33cfe5d4d0616e45b8d0"),
  "nome" : "Angelina",
  "telefone" : "9677-2323",
  "idade" : 49,
  "curso" : "EC",
  "inadimplente" : true,
  "uf" : "SP",
  "cidade" : "",
  "profissao" : "professora"
}
{
  "_id" : ObjectId("626f39a4e5d4d0616e45b8d4"),
  "nome" : "Bianca",
  "telefone" : "9677-2323",
  "idade" : 18,
  "curso" : "EC",
  "inadimplente" : true,
  "uf" : "SP",
  "cidade" : ""
}
```

db.clientesCad.insertOne({nome:"Luna"})

```
{
  "_id" : ObjectId("626f39c3e5d4d0616e45b8d5"),
  "nome" : "Juca",
  "telefone" : "9677-2323",
  "idade" : 18,
  "curso" : "EC",
  "inadimplente" : true,
  "uf" : "SP",
  "cidade" : ""
}
{ "_id" : ObjectId("626f3fd2e5d4d0616e45b8d6"), "nome" : "Luna" }
>
```

Excluindo

Para excluir pode ser usado o deleteOne() ou deleteMany()

```
> db.clientesCad.deleteOne({nome:"Luna"})  
{ "acknowledged" : true, "deletedCount" : 1 }
```

```
> db.clientesCad.deleteMany({idade:18})  
{ "acknowledged" : true, "deletedCount" : 2 }
```

Excluindo a collection

O drop() pode excluir uma collection ou um banco de dados inteiros.

```
db.clientesCad.drop()
```

```
db.dropDatabase()
```

Operadores

Operadores	Significado
\$gt	Maior que (greater-than)
\$gte	Igual ou Maior que (greater-than or equal to)
\$lt	Menor que (less-than)
\$lte	Igual ou Maior que (less-than or equal to)
\$ne	Não igual (not equal)
\$in	Existe em uma lista
\$nin	Não existe em uma lista
\$all	Existe em todos os elementos
\$not	Busca o contrário da condição
\$mod	Calcula o módulo
\$exists	Verifica se o campo existe
\$elemMatch	Compara elementos de array
\$size	Compara tamanho de array

Operadores

Recupera todos os documentos da coleção inventory em queo status é igual a "A"ou "D":

```
db.inventory.find( { status: { $in: [ "A", "D" ] } } )
```

```
> db.clientesCad.find({idade: {$gte:18}})
{ "_id" : ObjectId("626f33cfe5d4d0616e45b8d0"), "nome" : "Angelina", "telefone" : "9677-2323", "idade" : 49, "curso" : "EC", "inadimplente" : true, "uf" : "SP", "cidade" : "", "profissao" : "professora" }
{ "_id" : ObjectId("626fcb52e5d4d0616e45b8d7"), "nome" : "Bianca", "telefone" : "9677-2323", "idade" : 18, "curso" : "EC", "inadimplente" : false }
{ "_id" : ObjectId("626fcb70e5d4d0616e45b8d8"), "nome" : "Juca", "telefone" : "15 8677-2323", "idade" : 18, "curso" : "EP", "inadimplente" : false }
>
```

Operadores

```
> db.clientesCad.updateMany({idade: {$lte : 18} }, {$set : {inadimplente: false}})
{ "acknowledged" : true, "matchedCount" : 2, "modifiedCount" : 0 }
```

Acrescentei mais um campo para o cliente professor

db.clientesCad.updateMany({nome: "Angelina" }, {\$set : {salario: 2000}}) e depois incrementei mil a mais para o salário deste cliente.

```
> db.clientesCad.find({nome: "Angelina"})
{ "_id" : ObjectId("626f33cfe5d4d0616e45b8d0"), "nome" : "Angelina", "telefone" : "9677-2323", "idade" : 49, "curso" : "Educação Infantil", "inadimplente" : true, "uf" : "SP", "cidade" : "", "profissao" : "professora", "salario" : 2000 }
> db.clientesCad.updateOne({nome:"Angelina"}, {$inc: {salario: 1000}})
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.clientesCad.find({nome: "Angelina"})
{ "_id" : ObjectId("626f33cfe5d4d0616e45b8d0"), "nome" : "Angelina", "telefone" : "9677-2323", "idade" : 49, "curso" : "Educação Infantil", "inadimplente" : true, "uf" : "SP", "cidade" : "", "profissao" : "professora", "salario" : 3000 }
```

******Para decrementar colocaria o sinal de -, como -1000.

<https://medium.com/danieldiasjava/mongo-shell-749e21fe62c3>