

Formação Cientista de Dados

MongoDB

MongoDB

MongoDB



- Open source
- Multiplataforma
- Escalável
- Orientado a documentos: JSON
- Permite documentos aninhados
- Indexados: pode-se buscar o conteúdo dos documentos
- Não tem schema fixo
- Não tem integridade referencial

Estrutura

Relacional

Banco de Dados

Tabela

Linha

Coluna

MongoDB

Banco de Dados

Coleção

Documento

Campo

Principais Tipos de Dados



String



Inteiro



Booleano



Double



Array

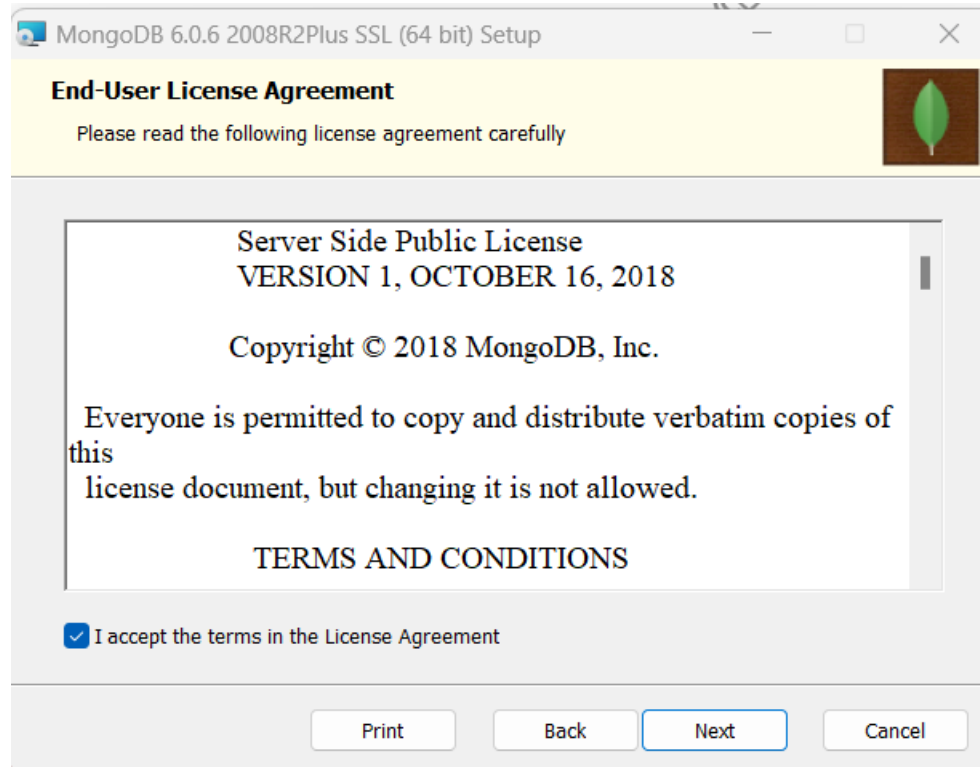


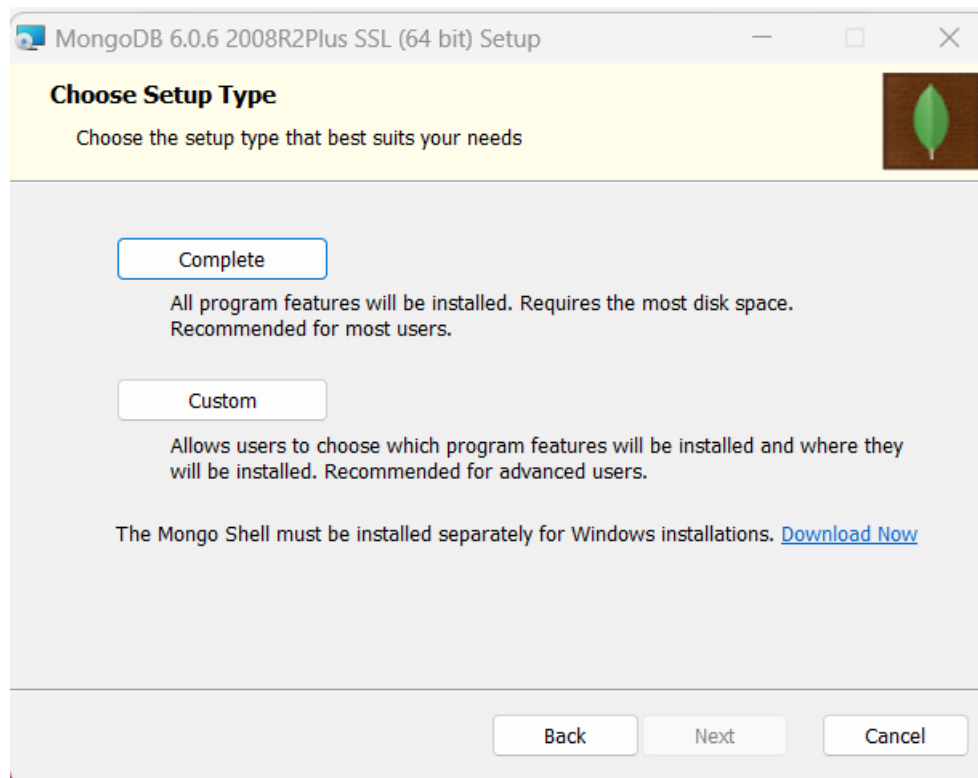
Timestamp



Object







MongoDB 6.0.6 2008R2Plus SSL (64 bit) Service Customizati...

Service Configuration

Specify optional settings to configure MongoDB as a service.

☒ Install MongoD as a Service

☒ Run service as Network Service user

☐ Run service as a local or domain user:

Account Domain:

Account Name:

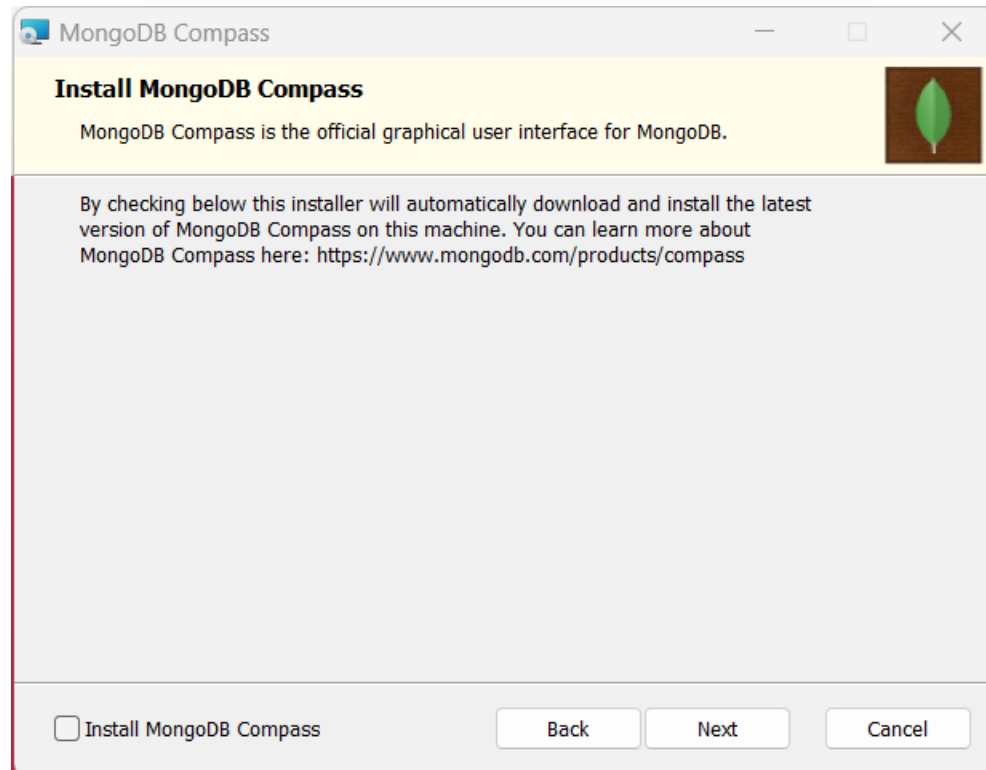
Account Password:

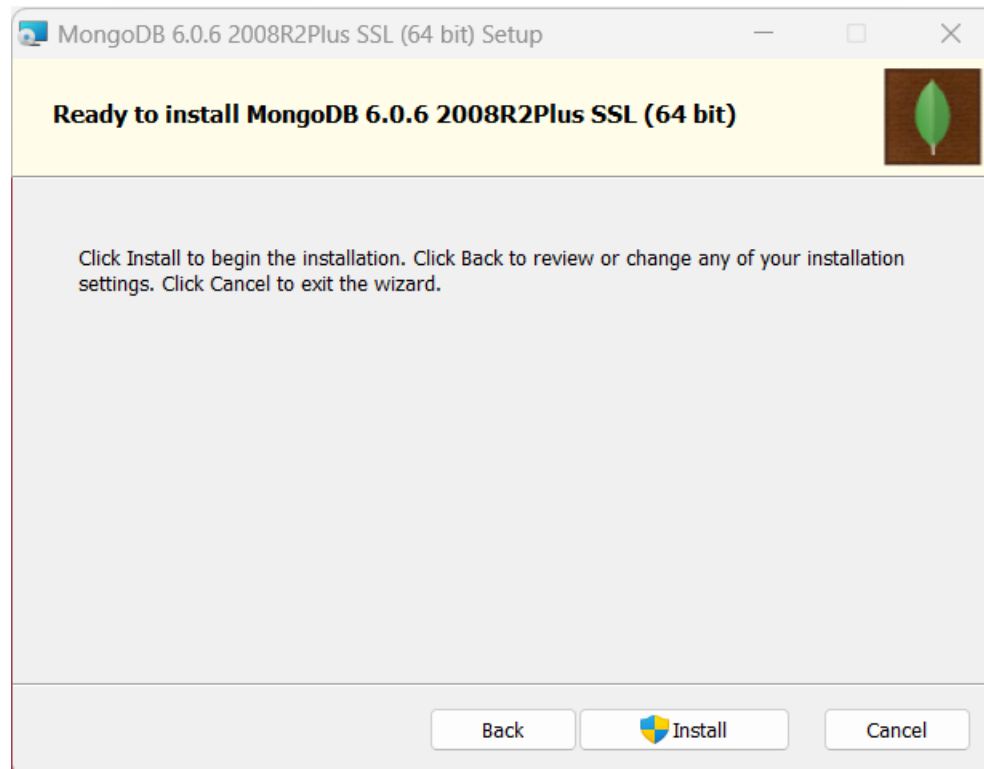
Service Name:

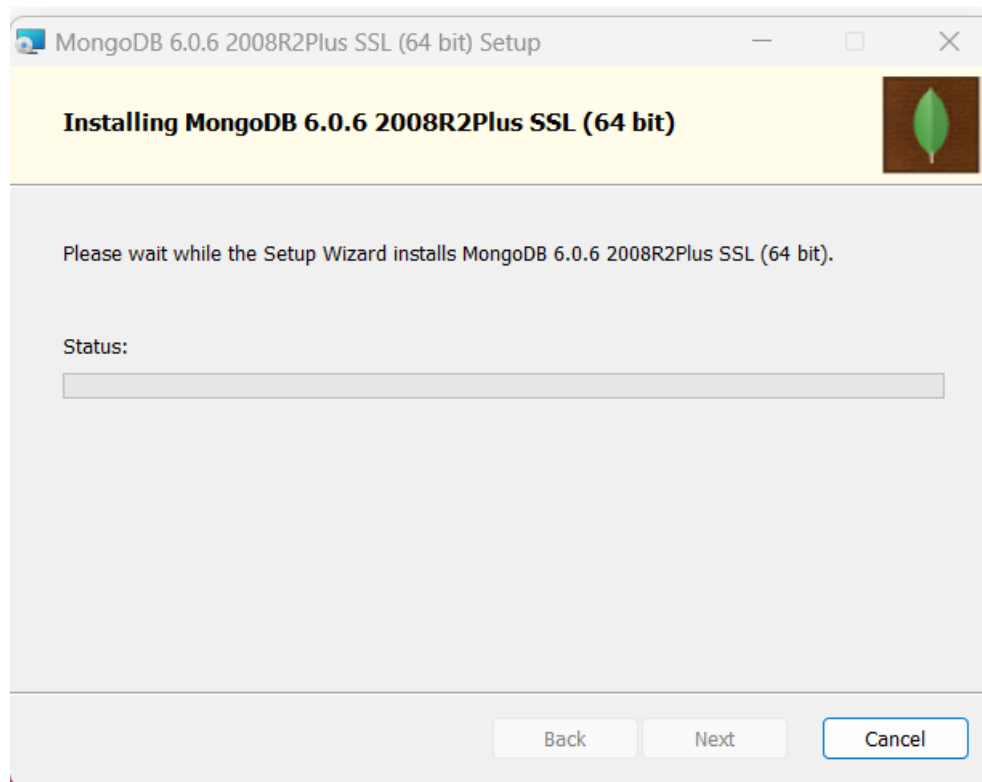
Data Directory:

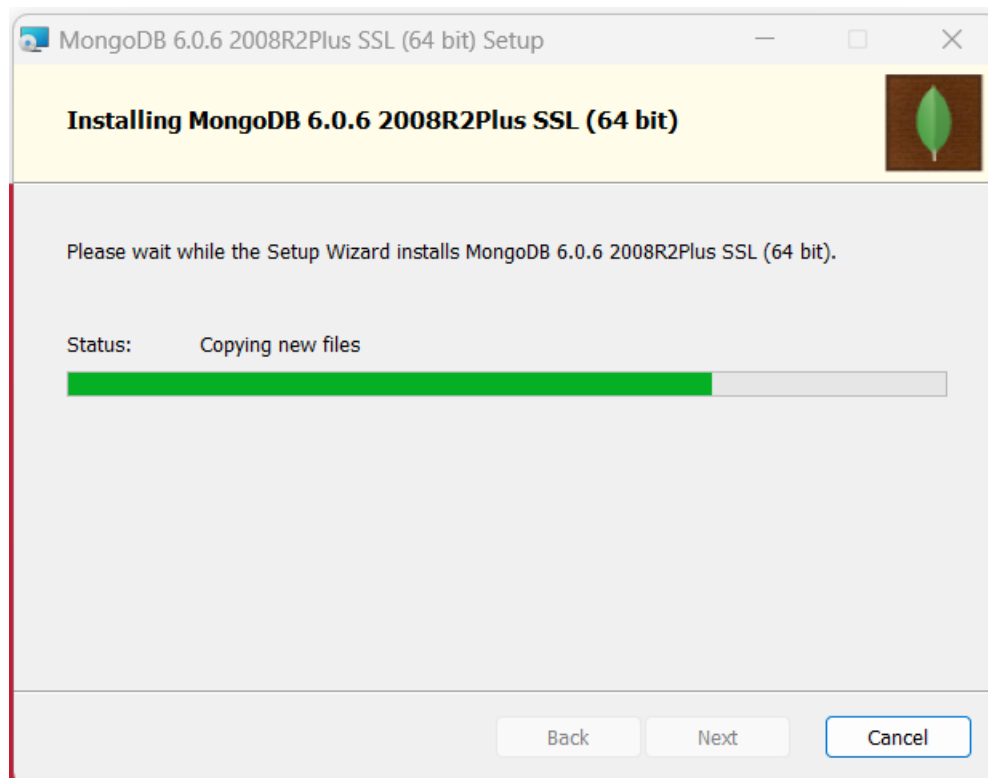
Log Directory:

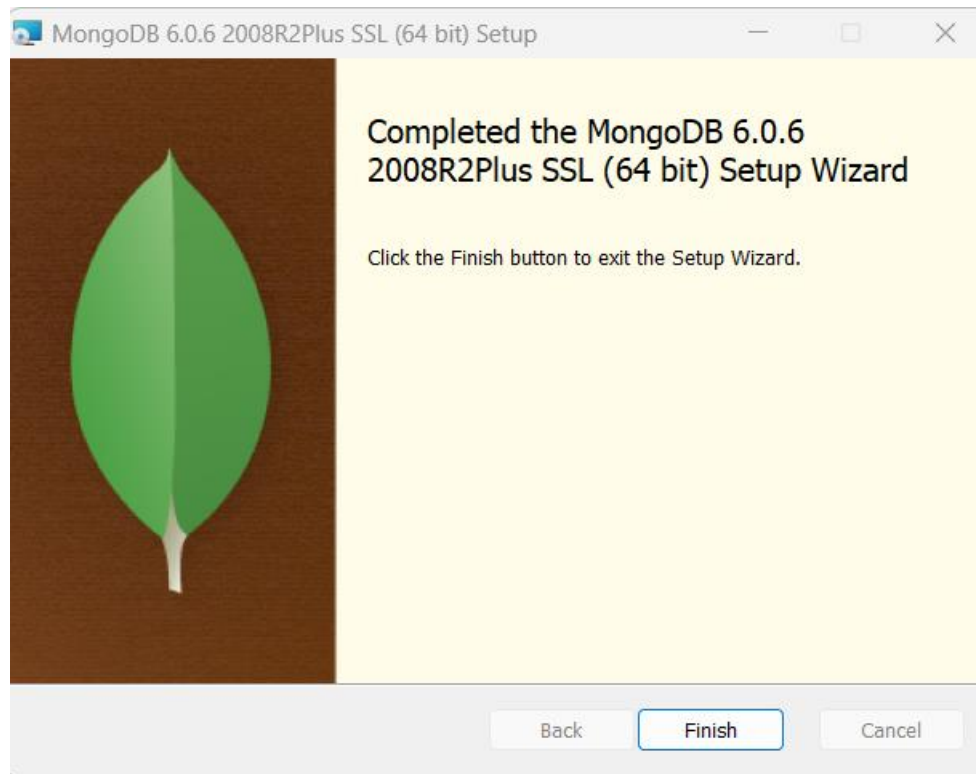
< Back Next > Cancel

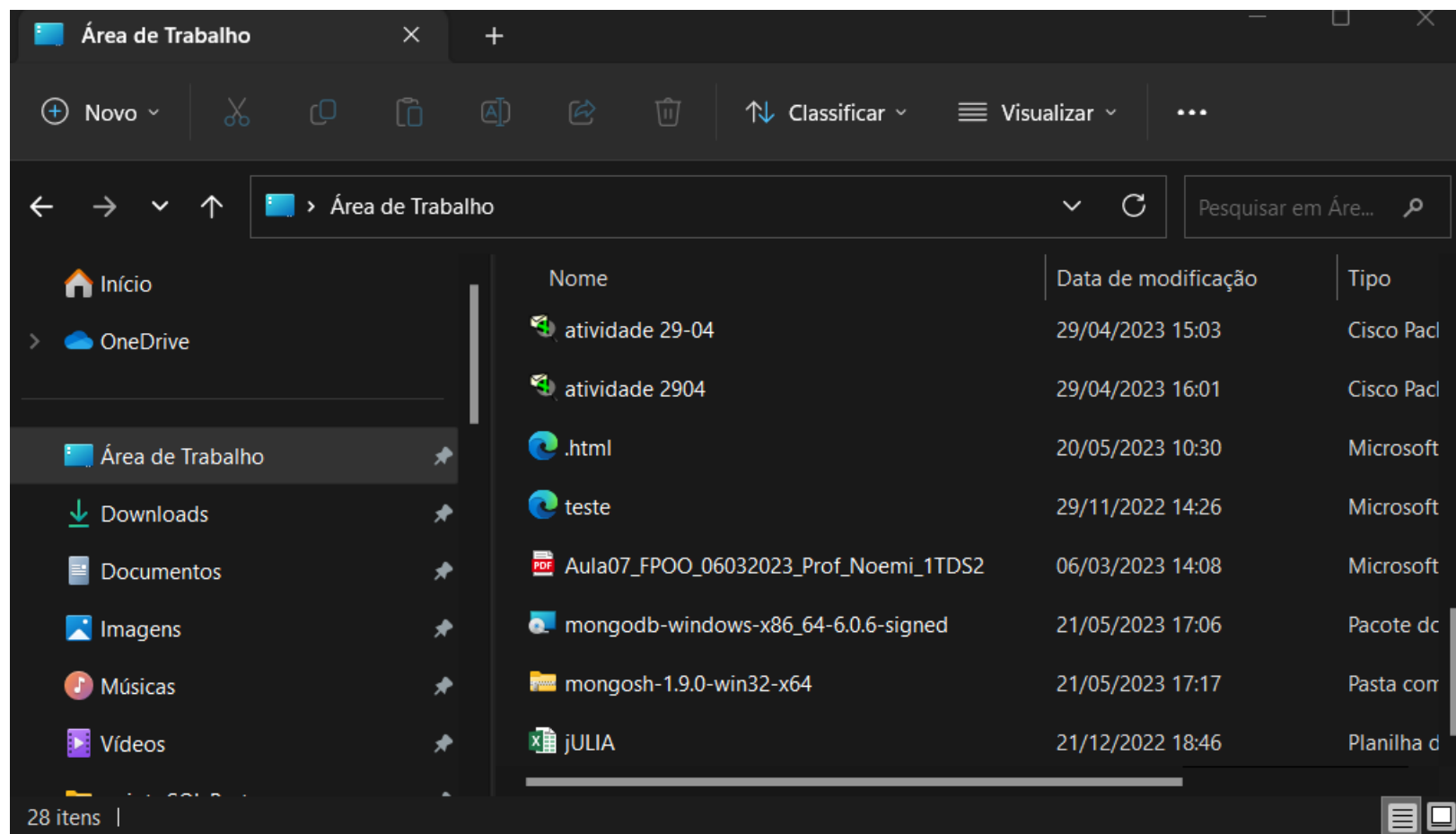




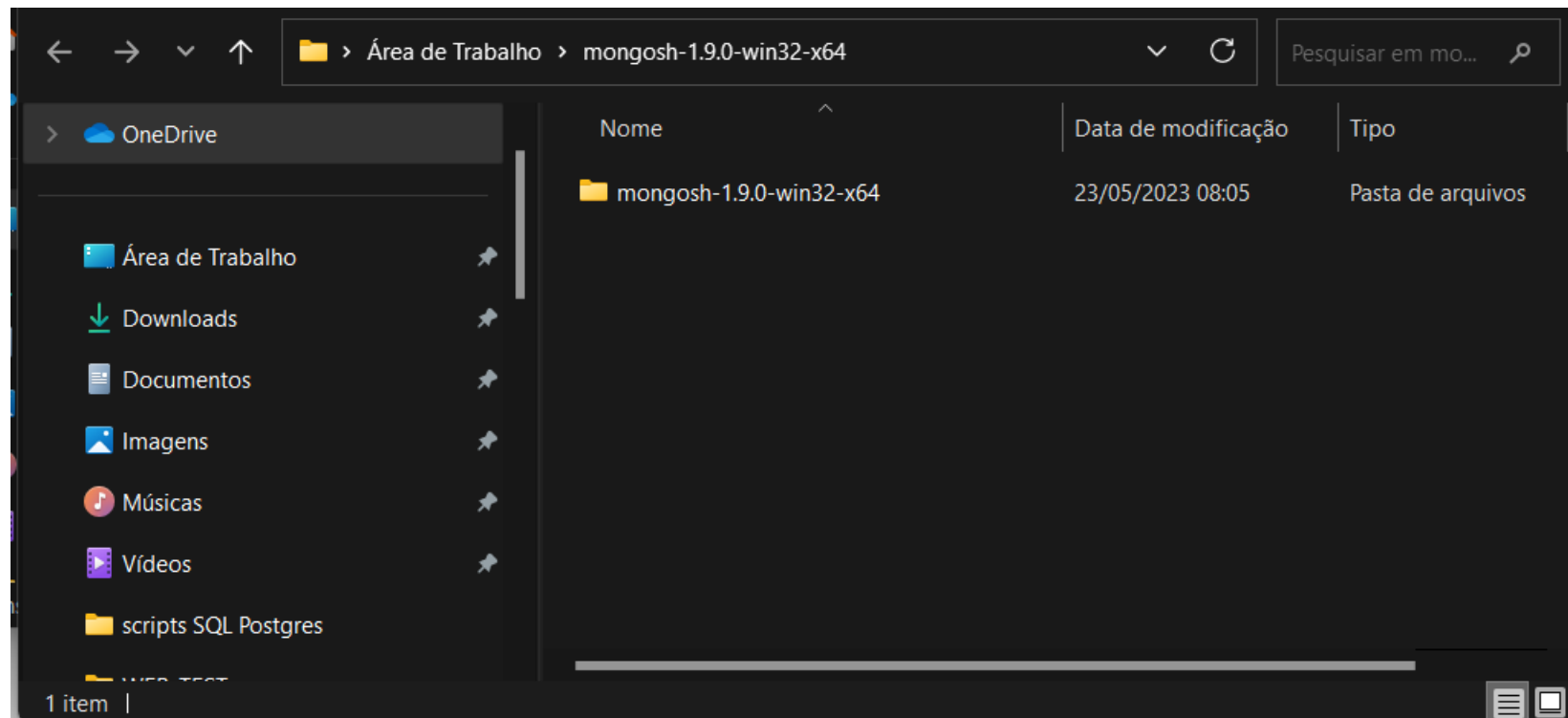


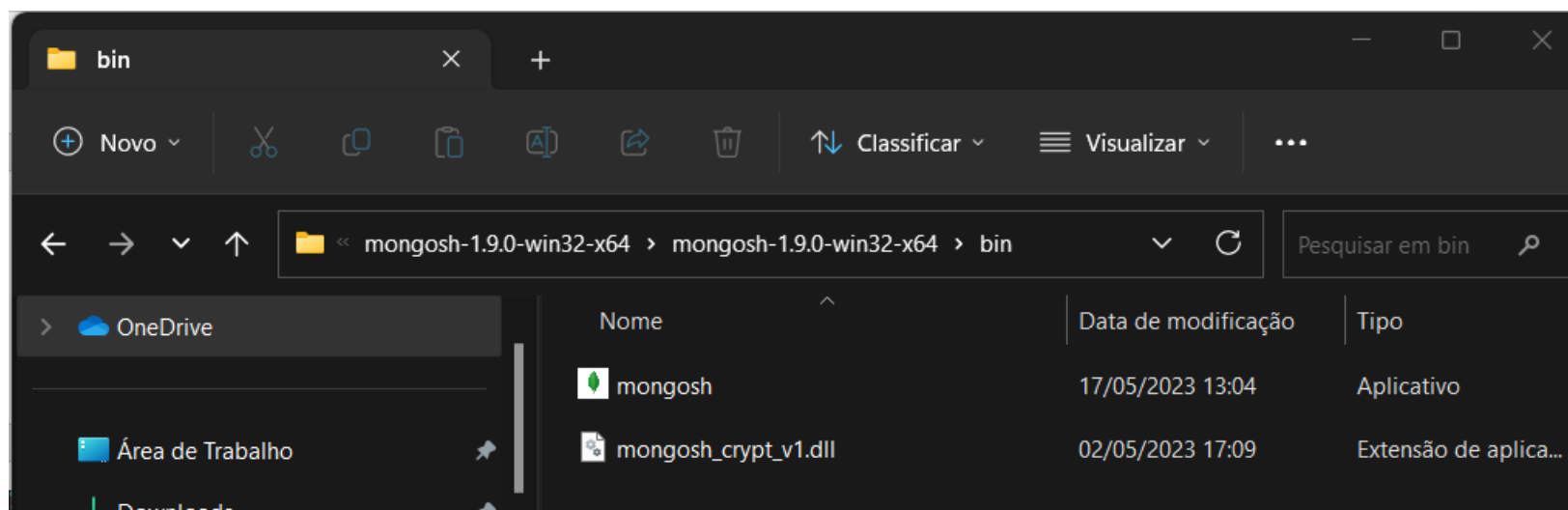


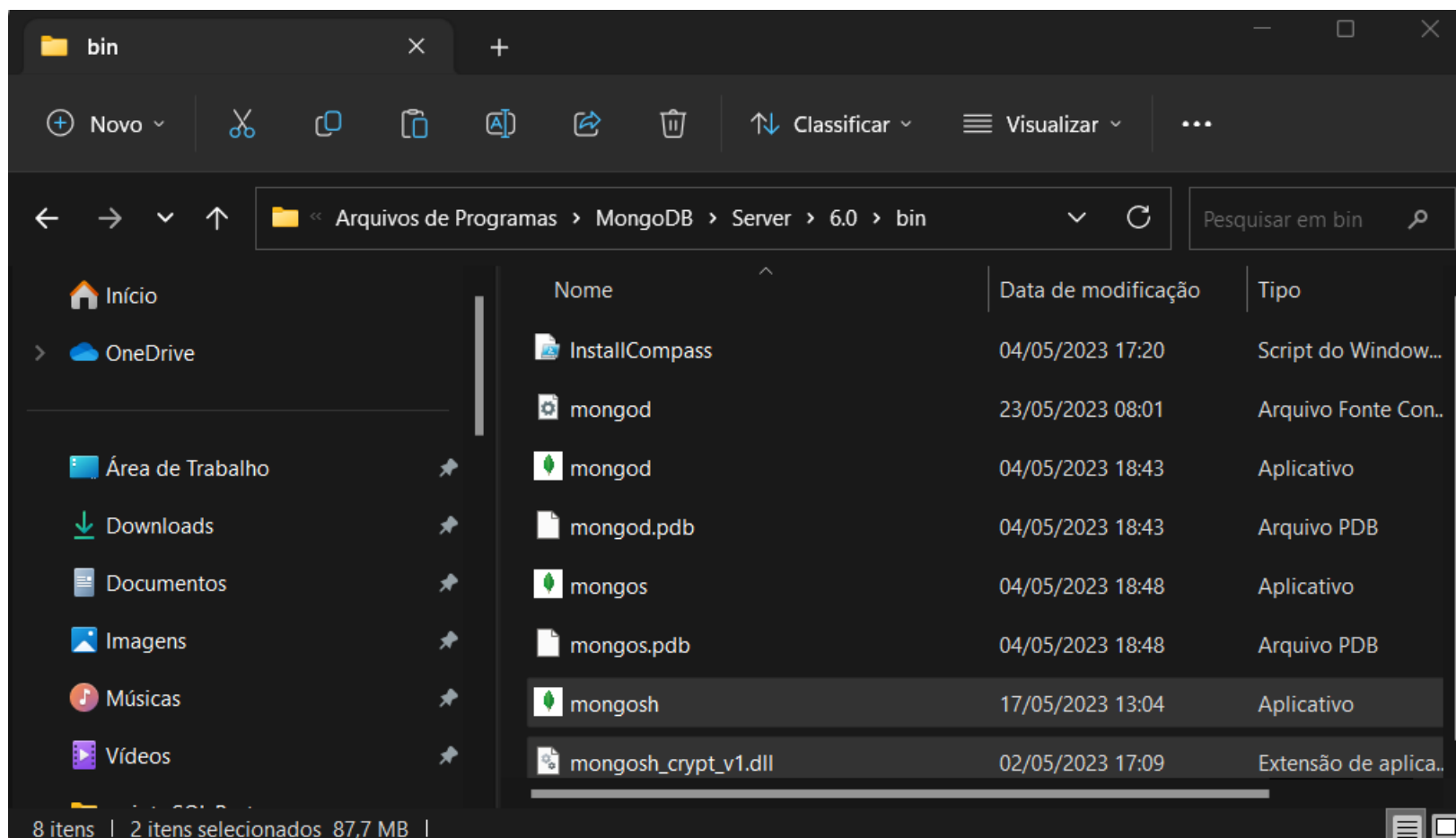




C:\Program Files\MongoDB\Server\6.0\bin







Anaconda Powershell Prompt

(base) PS C:\Users\senai>

Diretório: C:\Program Files\MongoDB\Server\6.0\bin

Mode	LastWriteTime		Length	Name
----	-----	-----	-----	----
-a----	04/05/2023	17:20	1558	InstallCompass.ps1
-a----	23/05/2023	08:01	605	mongod.cfg
-a----	04/05/2023	18:43	56496128	mongod.exe
-a----	04/05/2023	18:43	847228928	mongod.pdb
-a----	04/05/2023	18:48	34883072	mongos.exe
-a----	04/05/2023	18:48	474836992	mongos.pdb
-a----	17/05/2023	13:04	70952448	mongosh.exe
-a----	02/05/2023	17:09	21039616	mongosh_crypt_v1.dll

(base) PS C:\Program Files\MongoDB\Server\6.0\bin> **mongosh.exe**

```
(base) PS C:\Program Files\MongoDB\Server\6.0\bin> mongosh.exe
Current Mongosh Log ID: 646c9fb6712382a5ac844d30
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true
.9.0
Using MongoDB:      6.0.6
Using Mongosh:      1.9.0

For mongosh info see: https://docs.mongodb.com/mongodbs-shell/

To help improve our products, anonymous usage data is collected and sent to our
telemetry service (see https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

-----
The server generated these startup warnings when booting
2023-05-23T08:01:42.148-03:00: Access control is not enabled for the database.
Configuration is unrestricted
-----

test>
```

Comando para criar um BD

```
> use dbsenai
```

```
> db.posts.insert({nome:"Jose",postagem:"Bons Produtos!", data:"23-05-2023"})
```

```
> db.post.insert([ {nome:"Joao",postagem:" Pessimo Pagador!", data:"18-04-2023"}, {nome:"Maria",postagem:"Encontrei Bons Produtos!", data:"03-05-2023"}, {nome:"Celso",postagem:"Otimo atendimento!", data:"20-03-2023"} ])
```

```
test>

test> use dbsenai
switched to db dbsenai
dbsenai> db.posts.insert({nome:"José",postagem:"Bons Produtos!", data:"23-05-2023"})
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("646ca885712382a5ac844d31") }
}
dbsenai> db.posts.find()
[
  {
    _id: ObjectId("646ca885712382a5ac844d31"),
    nome: 'José',
    postagem: 'Bons Produtos!',
    data: '23-05-2023'
  }
]
dbsenai> |
```

> db.posts.find().pretty()

```
db.posts.find( ).pretty( )
```

```
db.posts.findOne( )
```

```
db.posts.find({nome:"Jose"})
```

```
db.posts.insert({nome:"Jose",postagem:"Loja suja!", data:"01-05-2023"})
```

```
db.posts.find({nome:"Jose",postagem:"Bons Produtos!"})
```

```
db.post.insert([ {nome:"André",postagem:" Produtos caros!", data:"12-01-2023", idade:25},  
{nome:"Ricardo",postagem:"Produtos caros!", data:"14-06-2023", idade:12},  
{nome:"Severino",postagem:"Bom atendimento!", data:"20-03-2023"} ])
```

```
db.posts.find({postagem:"Produtos caros", idade: {$lte: 12}})
```

```
db.posts.find({$or:[{nome:"Jose"},{nome: "Severino"}]}))
```

```
db.posts.find({postagem: { $in: ["Produtos caros", "Loja suja!]]}})
```

```
db.posts.find({nome: "Jose"},{_id: 0, nome: 0})
```

```
db.posts.find( ).limit(2)
```

```
db.posts.find( ).limit(3)
```

```
db.posts.find( ).sort({nome:1})
```

```
db.posts.find( ).sort({nome:-1})
```

```
db.posts.find( ).sort({nome:-1}).limit(2)
```



```
db.posts.find({nome: "Andre"})
```

```
db.posts.update({nome: "Andre"}, {$set: {idade: 29}})
```

```
db.post.deleteOne({nome: "Andre"})
```

```
db.post.find({nome: "Andre"})
```

```
db
```

```
show dbs
```

```
show collections
```

No powershell

Executar :

C:\programa Files\MongoDB\Sever\6.0\bin\mongoimport.exe --db escola –
Collection estudantes – file “c:\dados\students.json”

CONEXÃO DO PYTHON MONGODB

No powershell instalar pymongo:

```
# pip install pymongo
```

No jupyter

```
from pymongo import MongoClient
```

```
#Conexão local padrão
```

```
cliente = MongoClient('mongodb://localhost:27017/*')
```

```
#banco dbsenai
```

```
db = cliente.dbsenai
```

```
#coleção posts
```

```
Conexao = db.posts
```

```
#um documento
```

```
print(conexão.find_one())
```

```
print(conexão.find_one({"nome":"Maria"}))
```

```
#todos
```

```
for conexão in conexão.find():  
    print(conexão)
```

```
Cliente.close()
```