



# **DESENVOLVIMENTO DE SOFTWARE PARA WEB**

**Bruno da Silva Sales**

**MongoDB**

**QUIXADÁ**

**2022**

## 1. Popule a base de dados

1. Adicione 4 usuários à sua coleção usuários com nome “pedro”, “alice”, ”maria” e “joao”. Os outros atributos são de sua escolha

R:

### Pedro

```
> show dbs
admin    0.000GB
config  0.000GB
local    0.000GB
> var usuario = {nome:"Pedro", matricula:498877, dataNascimento:"05-08-2001"}
> usuario
{ "nome" : "Pedro", "matricula" : 498877, "dataNascimento" : "05-08-2001" }
> db.usuarios.insert(usuario)
WriteResult({ "nInserted" : 1 })
```

```
> var a = usuarios.findOne()
uncaught exception: ReferenceError: usuarios is not defined :
@(shell):1:5
> var a = db.usuarios.find()
> a.next()
{
  "_id" : ObjectId("627958646afac5e5cc0314ba"),
  "nome" : "Pedro",
  "matricula" : 498877,
  "dataNascimento" : "05-08-2001"
}
> |
```

### Alice

```
> var usuario = {nome:"Alice", matricula:889977, dataNascimento:"25-06-2002"}
> db.usuarios.insert(usuario)
WriteResult({ "nInserted" : 1 })
```

### Maria

```
> var a = {nome:"Maria", matricula:556611, dataNascimento:"22-06-2045"}
> db.usuarios.insert(a)
WriteResult({ "nInserted" : 1 })
```

### Joao

```
> db
escola
> var usuario = {nome:"Joao", matricula:774455, dataNascimento:"14-05-2014"}
> db.usuarios.insert(usuario)
WriteResult({ "nInserted" : 1 })
> var a = db.usuarios.findOne()
```

```

> var a = db.usuarios.find()
> a.next().next()
uncaught exception: TypeError: a.next(...).next is not a function :
@(shell):1:1
> a.next()
{
  "_id" : ObjectId("627959706afac5e5cc0314bb"),
  "nome" : "Alice",
  "matricula" : 889977,
  "dataNascimento" : "25-06-2002"
}
> |

```

obs: Tentei dois next() encadeados, para tentar pegar apenas o segundo valor logo, mas não dá

## 2. Elabore as seguintes consultas

### 1. Obtenha todos usuários que tenham mais de 25 anos

```

> var a = {idade: {"$gt":25}}
> db.usuarios.find(a)
{ "_id" : ObjectId("62795b7e6afac5e5cc0314be"), "nome" : "Bruno", "idade" : 36 }
> db.usuarios.find(a).pretty()
{
  "_id" : ObjectId("62795b7e6afac5e5cc0314be"),
  "nome" : "Bruno",
  "idade" : 36
}
> |

```

### 2. Obtenha um usuário tem nome igual a “joao”

```

> var a = {nome: {"$eq":"Joao"}}
> db.usuarios.find(a).pretty()
{
  "_id" : ObjectId("627959df6afac5e5cc0314bc"),
  "nome" : "Joao",
  "matricula" : 774455,
  "dataNascimento" : "14-05-2014"
}
> |

```

### 3. Obtenha todos os usuários que tem nome igual a “joao” e idade igual a 25 anos

Mude o banco porque estou fazendo em outro dia e pretendo praticar

```
> var usuario = {nome:"Joao", idade:25}
> show collections
> db.pessoas.insert(usuario)
WriteResult({ "nInserted" : 1 })
> var usuario = {nome:"Matheus", idade:25}
> db.pessoas.insert(usuario)
WriteResult({ "nInserted" : 1 })
> var usuario = {nome:"Joao", idade:24}
> db.pessoas.insert(usuario)
WriteResult({ "nInserted" : 1 })
> var criterio = {"$and":[{"nome":"Joao"},{idade:25}]}
> var.pessoas.find(criterio)
uncaught exception: SyntaxError: missing variable name :
@(shell):1:3
> db.pessoas.find(criterio)
{ "_id" : ObjectId("627a4ed9f7c0fdabe8b454be"), "nome" : "Joao", "idade" : 25 }
>
```

4. Obtenha todos os usuários que têm idade maior que 26 anos e nome diferente de “joao”

```
> var criterio = {"$and":[{"nome":{"$ne":"Joao"}},{idade: { "$gte" : 26}}]}
> db.pessoas.find(criterio)
> var usuario = {nome:"BrunoSales", idade:27}
> db.pessoas.insert(usuario)
WriteResult({ "nInserted" : 1 })
> db.pessoas.find(criterio)
{ "_id" : ObjectId("627a508df7c0fdabe8b454c1"), "nome" : "BrunoSales", "idade" : 27 }
> |
```

5. Obtenha todos os usuários que tem nome “joao” ou (idade maior que 25 e nome igual a “pedro”)

```
> var criterio = {"$or":[{"nome":{"$eq":"Joao"}},{ "$and": [ { idade: { "$gte" : 25 } }, { nome : {"$eq" : "Pedro"} } ] ] }
> db.pessoas.find(criterio)
{ "_id" : ObjectId("627a4ed9f7c0fdabe8b454be"), "nome" : "Joao", "idade" : 25 }
{ "_id" : ObjectId("627a4efff7c0fdabe8b454c0"), "nome" : "Joao", "idade" : 24 }
> |
```