



Banco de Dados Não Relacionais

NoSQL

Outros métodos utilizados junto com o find()



sort()

- ▶ O método `sort` é usado para ordenar os documentos em uma coleção com base em um ou mais campos.
- ▶ Você pode especificar a ordem de classificação como ascendente (1) ou descendente (-1) para cada campo. Isso permite que você recupere os documentos em uma ordem específica.

- ▶ Exemplo:

```
db.municipios.find().sort({codigo_uf: 1, nome: -1})
```

- ▶

count()

- ▶ O método **count** no MongoDB é usado para contar o número de documentos que correspondem a um determinado critério de consulta em uma coleção.
- ▶ Isso ajuda a determinar quantos documentos atendem a uma condição específica.

- ▶ Exemplo:

```
db.municipios.find({capital:true}).count()
```

- ▶

limit()

- ▶ O método `limit` é utilizado para limitar o número de documentos retornados por uma consulta.
- ▶ Ele restringe o número total de documentos que serão exibidos no resultado da consulta, o que pode ser útil para a paginação de resultados.

- ▶ Exemplo:

```
db.municipios.find({capital:true}).limit(5)
```



skip()

- ▶ O método **skip** é usado para pular um certo número de documentos na coleção.
- ▶ Isso é frequentemente usado em conjunto com o método **limit** para implementar a paginação, permitindo que você selecione um conjunto específico de documentos, ignorando os primeiros documentos.

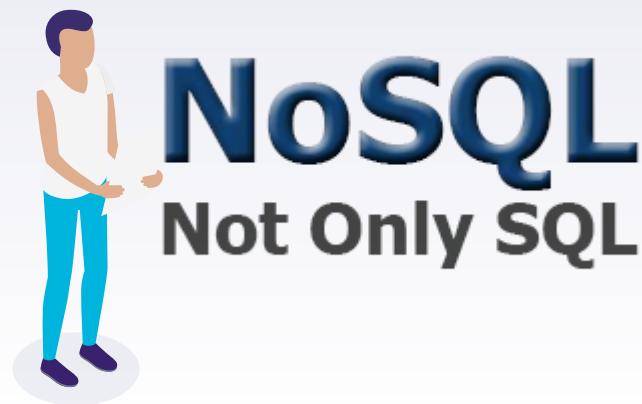
▶

- ▶ Exemplo:

```
db.municipios.find({capital:true}).limit(5).skip(5)
```

NoSQL

Buscas parciais em strings no método find



Insensitive case

- ▶ Encontrar todos os documentos com o campo "nome" igual a "João", independente se está em maiúsculo ou minúsculo

- ▶ Exemplo:

```
db.collection.find({nome: /João/i})
```

- ▶

Que inicie com algo ^

- ▶ Encontrar todos os documentos com o campo "email" que começam com "contato":

- ▶ Exemplo:

```
db.collection.find({email: /^contato/})
```

Que termine com algo \$

- ▶ Encontrar todos os documentos com o campo "nome" que terminem com "Silva":

- ▶ Exemplo:

```
db.collection.find({nome: /Silva$/})
```

Operador \$regex

- ▶ O operador \$regex no MongoDB permite realizar buscas por padrões de texto dentro de strings. É uma ferramenta poderosa para filtrar documentos com base em critérios específicos.

- ▶ Exemplo:

```
db.collection.find({nome: {$regex: /^Ssilva/i}})
```

Operador \$regex

- ▶ Encontrar todos os documentos com o campo "nome" que contém a palavra "Silva" ignorando maiúsculas e minúsculas e que começa com a letra "S":

- ▶ Exemplo:

```
db.collection.find({nome: {$regex: /^Ssilva/i}})
```

Operador \$regex

- ▶ Encontrar todas as pessoas com o nome Luis, Luiz, Luís ou Luíz.

- ▶ Exemplo:

```
db.collection.find({ nome: { $regex: /^Lu[ií][sz]$/i } })
```