

Formação Cientista de Dados

MongoDB



Exemplo

- Criar banco de dados com comentários de usuários coletados em redes sociais

Cria Banco de Dados

- Comando "use" acessa banco
- Acessar banco inexistente cria o banco
 - É necessário inserir dados para persistir o banco de dados

> use dbmidias

switched to db dbmidias



Inserir

- **Insert**

- Insere um único documento na coleção posts

```
>db.posts.insert({nome:"José",postagem:"Bons  
Produtos!", data:"31-06-2019"})  
WriteResult({ "nInserted" : 1 })
```

Acima a coleção é criada implicitamente

- Para criar a coleção primeiro:

```
>db.createCollection("clientes")
```



Inserir Vários Documentos

```
db.posts.insert([
  {nome:"Antonio",postagem:"Minha bike quebrou", data:"26-05-2019"},
  {nome:"Maria Silva",postagem:"Encontrei tudo que procurava", data:"14-06-2019"},
  {nome:"Lucas Andrade",postagem:"Ótimo atendimento!", data:"12-04-2019"}
])

BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 3,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})
```



Recuperar Documentos



`db.posts.find()`



`db.posts.findOne()`



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



Ler Documentos: find

```
> db.posts.find()

{ "_id" : ObjectId("5d090bc10ee1100c307004d4"),
  "nome" : "José", "postagem" : "Bons Produtos!",
  "data" : "31-06-2019" }

{ "_id" : ObjectId("5d090cd10ee1100c307004d5"),
  "nome" : "Antonio", "postagem" : "Minha bike
quebrou", "data" : "26-05-2019" }

{ "_id" : ObjectId("5d090cd10ee1100c307004d6"),
  "nome" : "Maria Silva", "postagem" : "Encontrei
tudo que procurava", "data" : "14-06-2019" }

{ "_id" : ObjectId("5d090cd10ee1100c307004d7"),
  "nome" : "Lucas Andrade", "postagem" : "Ótimo
atendimento!", "data" : "12-04-2019" }
```



Ler Documentos: pretty

```
> db.posts.find().pretty()  
{  
  "_id" : ObjectId("5d090bc10ee1100c307004d4"),  
  "nome" : "José",  
  "postagem" : "Bons Produtos!",  
  "data" : "31-06-2019"  
}
```



Ler Documentos: findOne

```
> db.posts.findOne()  
{  
  "_id" : ObjectId("5d01129c1dd51b69599c3110"),  
  "nome" : "José",  
  "postagem" : "Bons Produtos",  
  "data" : "31-06-2019"  
}
```



Ler Documentos Específico

```
> db.posts.find({nome:"José"})  
{ "_id" : ObjectId("5d090bc10ee1100c307004d4"),  
  "nome" : "José", " Postagem" : "Bons Produtos!",  
  "data" : "31-06-2019" }
```



Condição AND

```
> db.posts.insert({nome:"José",postagem:"Loja  
suja!", data:"30-01-2019"})
```

```
WriteResult({ "nInserted" : 1 })
```

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

```
{ "_id" : ObjectId("5d090bc10ee1100c307004d4"),  
  "nome" : "José", "postagem" : "Bons Produtos!",  
  "data" : "31-06-2019" }
```



Operadores

- \$eq: igual
- \$gt: maior que
- \$gte: maior ou igual que
- \$lt: menor que
- \$lte: menor ou igual que
- \$ne: diferente de
- \$in: contém
- \$nin: não contém



Condição AND

```
db.posts.insert([  
  {nome:"André",postagem:"Produtos caros", data:"12-01-2019",idade:25},  
  {nome:"Ricardo",postagem:"Produtos caros", data:"14-07-2019", idade:12}])
```

#idade menor ou igual a 12

```
> db.posts.find({postagem:"Produtos caros",idade: {$lte: 12}})  
{ "_id" : ObjectId("5d0911600ee1100c307004da"), "nome" : "Ricardo",  
  "postagem" : "Produtos caros", "data" : "14-07-2019", "idade" : 12 }
```



Condição OR

```
> db.posts.find({$or:[{nome:"José"},{nome: "Antonio"}]})  
{ "_id" : ObjectId("5d090bc10ee1100c307004d4"), "nome" : "José",  
  " Postagem" : "Bons Produtos!", "data" : "31-06-2019" }  
{ "_id" : ObjectId("5d090cd10ee1100c307004d5"), "nome" : "Antonio",  
  " Postagem" : "Minha bike quebrou", "data" : "26-05-2019" }  
{ "_id" : ObjectId("5d090ee00ee1100c307004d8"), "nome" : "José",  
  " Postagem" : "Loja suja!", "data" : "30-01-2019" }
```

