

Banco de Dados NoSQL **- Banco de Dados de** **Documentos**

Prof. Henrique Batista da Silva

Importando de um
arquivo .json
Airbnb dataset

Banco de Dados de Documentos

A biblioteca pymongo do Python é utilizado para conectar um programa em Python com o MongoDB (banco de dados baseado em documentos).

Veja o código abaixo (também disponível no slide de aula)

```
import pymongo
import json

client = pymongo.MongoClient("localhost", 27017)
db = client.discografia

json_string = '{"nome" : "Somewhere Far Beyond", "dataLancamento" : "1992-05-30", "duracao" : "3328", "artista" : {"nome" : "Blind Guardian"}}'
album = json.loads(json_string)

db.albums.insert(album)
```

Observe que neste exemplo utilizamos a biblioteca json do Python para criar um documento e inserir no banco de dados. O nome do database é "discografia" e o nome da collection é "albums".

Para consultar o nome do artista que gravou o álbum "Somewhere Far Beyond":

```
import pymongo
client = pymongo.MongoClient("localhost", 27017)
db = client.discografia

album = db.albums.find_one({"nome" : "Somewhere Far Beyond"})

nome = album["artista"]["nome"]
print(nome)
```

Além de usar o Python você também pode escrever suas consultas diretamente em Javascript pelo cliente do MongoDB

Importando arquivo .csv para o MongoDB (versão Docker Linux):

Para importar uma base de dados obtida da internet diretamente para o MongoDB, usamos o "mongoimport"

Se estiver usando Docker, deve-se realizar o seguinte procedimento:

1 – Faça download do dataset no seguinte link (host Linux)

https://raw.githubusercontent.com/hortonworks/data-tutorials/master/tutorials/hdp/beginners-guide-to-apache-pig/assets/driver_data.zip

descompactar o arquivo .zip em um diretório qualquer e encontrará três arquivos no formato .csv

2 - Execute o Docker do MongoDB:

```
sudo docker start mongo
```

3 – Entre no cliente do Mongo e crie um banco no MongoDB chamado "teste" (veja como criar um novo banco no material de aula)

```
sudo docker run -it --link mongo:mongo --rm mongo mongo --host mongo test
```

4 – Execute o seguinte comando para copiar o arquivo .csv para dentro do Docker:

```
sudo docker cp /home/nosql/Downloads/driver_data/drivers.csv  
mongo:/tmp/drivers.csv
```

5 – Agora execute o comando "mongoimport" dentro do docker

```
sudo docker exec mongo mongoimport -d teste -c driver --type csv --headerline  
--file /tmp/drivers.csv
```

6 – Retorne o cliente do MongoDB e veja que os documentos foram criados com sucesso. Faça uma consulta para se certificar.

7 – Repita os passos 4 a 6 para os outros dois arquivos.

Este procedimento também funciona para arquivos .json. Se um Docker não estiver sendo usado, ou seja, se estiver conectado diretamente em uma máquina com o MongoDB instalado, não será necessário realizar as etapas 2, 3 e 4. Além disso, na etapa 5 o comando é mais simples, pois não precisa envolver o Docker:

```
mongoimport -d teste -c driver --type csv --headerline --file /tmp/drivers.csv
```

Importando arquivo .csv para o MongoDB (versão Windows):

Para importar uma base de dados obtida da internet diretamente para o MongoDB, usamos o “mongoimport”

1 – Faça download do dataset no seguinte link

https://raw.githubusercontent.com/hortonworks/data-tutorials/master/tutorials/hdp/beginners-guide-to-apache-pig/assets/driver_data.zip

descompactar o arquivo .zip em um diretório qualquer (exemplo “c:\tmp”) e encontrará três arquivos no formato .csv

2 - Execute o MongoDB (veja o material de aula):

3 – Agora execute o comando “mongoimport”

```
.\mongoimport.exe --db=teste --collection=driver --type=csv --headerline --file=C:\tmp\drivers.csv
```

4 – Retorne o cliente do MongoDB e veja que os documentos foram criados com sucesso (em um database chamado “teste”). Faça uma consulta para se certificar.

5 – Repita os passos 3 e 4 para os outros dois arquivos.

Este procedimento também funciona para arquivos .json. Neste caso, faça:

```
.\mongoimport.exe --db=teste --collection=driver --type=json --  
file=C:\tmp\filename.json
```

Faça algumas consultas nas coleções para testar o procedimento.

Observação:

A partir da versão 4.4 do MongoDB o mongoimport deve ser instalado separadamente.

Baixe o MongoDB Database Tools e descompacte os arquivos dentro da pasta "bin" no diretório de instalação do MongoDB. Para copiar estes arquivos, abra o prompt de comando do Windows como administrador e utilize o comando de copiar do cmd

https://www.mongodb.com/try/download/database-tools?tck=docs_databasetools

Exercícios:

Faça download do dataset do AirBnB:

https://github.com/neelabalan/mongodb-sample-dataset/blob/main/sample_airbnb/listingsAndReviews.json

Importe o dataset para o MongoDB conforme o tutorial acima e implemente em Python um pequeno programa para exibir o resultado de consultas de quartos por palavras chaves dos reviews do usuário.

Exemplo 1: Encontre quartos que possuam em seus reviews a palavra "perfect"

```
db.airbnb.find({"reviews.comments" : { $regex : /perfect/ } }).pretty()
```

Exemplo 2: Encontre quartos que possuam em seus reviews frases como "best place to be ", "wonderful host" e "beautiful place".

```
db.airbnb.find( {$and : [ {"reviews.comments" : { $regex : /best place to be/ } }, {"reviews.comments" : { $regex : /wonderful host/} }, {"reviews.comments" : { $regex : /beautiful place/ } } ] } ).pretty()
```

Exemplo 3: Encontre quartos que possuam em seus reviews pelo menos uma das frases como "best place to be ", "wonderful host" e "beautiful place".

```
db.airbnb.find( {$or : [ {"reviews.comments" : { $regex : /best place to be/ } }, {"reviews.comments" : { $regex : /wonderful host/ } }, {"reviews.comments" : { $regex : /beautiful place/ } } ] } ).pretty()
```

Exemplo 4: Encontre quartos que possuam em seus reviews a frase como "best place to be "

```
db.airbnb.find( {"reviews" : {$elemMatch : {"comments" : { $regex : /best place to be/ } } } } ).pretty()
```

Exemplo 5: supressão de campos de consulta: agora retorna apenas os campos desejados.

```
db.airbnb.find( {"reviews" : {$elemMatch : {"comments" : { $regex : /best place to be/ } } } }, {address: 1, price: 1, "reviews.comments" : 1, _id : 0} ).pretty()
```

Você pode filtrar os quartos pelos reviews e pelo endereço. Faça pelo menos três consultas com frases ou palavras que você considera importante nos reviews e filtre estes quartos por cidade ou país desejado.

Dica: combine as consultas dos exemplos para montar suas consultas e veja o exemplo de como usar a biblioteca pymongo no tutorial acima.