

Engenharia de Dados

Introdução ao MongoDB

DCA0132 - Engenharia de Dados

Prof. Carlos M. D. Viegas

DCA

Departamento de Engenharia de Computação e Automação
Universidade Federal do Rio Grande do Norte

UFRN
UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

Introdução ao MongoDB

- O que é o MongoDB?
 - É um Sistema Gerenciador de Banco de Dados de código aberto e orientado a documentos no modelo NoSQL

```
{  
  name: "sue",  
  age: 26,  
  status: "A",  
  groups: [ "news", "sports" ]  
}
```

← field: value
← field: value
← field: value
← field: value

- Alguns de seus diferenciais são:
 - Alto desempenho: documentos embutidos e índices atuando sobre eles;
 - Rica linguagem de consulta: permite operações CRUD, agregações de dados, busca por texto e consultas geo-espaciais;
 - Alta disponibilidade: *replica set*;
 - Escalabilidade horizontal: *sharding*.



Introdução ao MongoDB

- Quão utilizado é o MongoDB?
 - <https://db-engines.com/en/ranking>

Rank			DBMS	Database Model	Score		
Nov 2022	Oct 2022	Nov 2021			Nov 2022	Oct 2022	Nov 2021
1.	1.	1.	Oracle +	Relational, Multi-model i	1241.69	+5.32	-31.04
2.	2.	2.	MySQL +	Relational, Multi-model i	1205.54	+0.17	-5.98
3.	3.	3.	Microsoft SQL Server +	Relational, Multi-model i	912.51	-12.17	-41.78
4.	4.	4.	PostgreSQL +	Relational, Multi-model i	623.16	+0.44	+25.88
5.	5.	5.	MongoDB +	Document, Multi-model i	477.90	-8.33	-9.45
6.	6.	6.	Redis +	Key-value, Multi-model i	182.05	-1.33	+10.55
7.	7.	↑ 8.	Elasticsearch	Search engine, Multi-model i	150.32	-0.74	-8.76
8.	8.	↓ 7.	IBM Db2	Relational, Multi-model i	149.56	-0.10	-17.96
9.	9.	↑ 11.	Microsoft Access	Relational	135.03	-3.14	+15.79
10.	10.	↓ 9.	SQLite +	Relational	134.63	-3.17	+4.83

Introdução ao MongoDB

- Quanto utilizado é o MongoDB?
 - <https://db-engines.com/en/ranking>
- Critérios utilizados no ranking DB-Engines
 - Menções do SGBD em mecanismos de busca;
 - Interesse geral no SGBD (Google Trends);
 - Frequência de discussões técnicas sobre o SGBD (Stack Overflow e DBA Stack Exchange);
 - Número de ofertas de emprego relacionadas ao SGBD;
 - Número de perfis em redes profissionais onde o SGBD é mencionado (Linkedin e Upwork);
 - Relevância em redes sociais.

Introdução ao MongoDB

- Instalando e Executando o MongoDB
 - O MongoDB possui versões para os sistemas Unix e Windows
 - <https://www.mongodb.com/docs/manual/installation/>
 - Para executar o MongoDB é necessário inicialmente executar o *daemon mongod* que será o servidor
 - Após o *daemon* estar em execução um cliente *mongo* ou *pymongo* pode acessar e interagir com o SGDB

Criando um Banco de Dados MongoDB

- O MongoDB abstrai diversos comandos DDL
 - Estruturas são criadas conforme estas se tornam necessárias
- Para criar um banco de dados, basta você usar o comando para acessar um banco que ainda não existe
 - Assim que um registro for inserido neste banco, ele será criado e persistido automaticamente

SQL → `CREATE DATABASE db_engdados;`

MONGO → `use db_engdados`

Criando um Banco de Dados MongoDB

- Coleções e Documentos (Insert)
 - O MongoDB organiza os dados em coleções de documentos
 - Cada documento possui um atributo identificador (`_id`) e uma quantidade qualquer de outros atributos
 - Não é necessário especificar o tipo dos atributos
 - Documentos diferentes em uma mesma coleção podem ter atributos diferentes
 - Para criar uma coleção, basta inserir um documento na mesma ou invocar `createCollection()`
 - Existem duas operações de inserção no MongoDB:
 - Inserção de um único documento: `insertOne()`
 - Recebe como parâmetro um único documento
 - Inserção de múltiplos documentos: `insertMany()`
 - Recebe como parâmetro um vetor de documentos

Criando um Banco de Dados MongoDB

- Coleções e Documentos (Insert)

SQL

```
db_engdados=#  
CREATE TABLE alunos (  
  id int not null,  
  nome varchar(30) not null,  
  curso varchar(30) not null  
);  
db_engdados=#  
INSERT INTO alunos  
VALUES (1, 'Maria', 'C&T')  
       (2, 'João', 'Computação');
```

MongoDB

```
db_engdados.alunos.insertOne({"_id"  
": 1, "nome": "Maria", "curso":  
"C&T"})  
  
db_engdados.alunos.insertMany([  
{"_id": 1, "nome": "Maria",  
"curso": "C&T"}, {"_id": 2,  
"nome": "João", "curso":  
"Computação"} ])
```


Criando um Banco de Dados MongoDB

- Coleções e Documentos (Select)
 - O MongoDB possui dois métodos principais para retornar informações de documentos
 - O método `find()` retorna um ponteiro para todos os documentos que atendem aos critérios especificados
 - O método `findOne()` retorna um único documento que atende aos critérios especificados
 - Caso exista mais de um documento atendendo aos critérios, o método `findOne()` retorna apenas o primeiro

SQL

```
SELECT * FROM alunos;
```

MongoDB

```
db.alunos.findOne()  
db.alunos.find()
```

Criando um Banco de Dados MongoDB

- Coleções e Documentos (Select)

- Ambos os métodos `find()` e `findOne()` permitem especificar, da mesma forma, critérios de seleção e projeção para o resultado

SQL

```
SELECT nome, curso FROM alunos WHERE nome = "Maria";
```

PROJEÇÃO

SELEÇÃO

MongoDB

```
db.alunos.find({"nome": "Maria"}, {"_id": 0, "nome": 1, "curso": 1})
```

SELEÇÃO

PROJEÇÃO

Criando um Banco de Dados MongoDB

- Coleções e Documentos (Update)
 - O MongoDB possui três métodos para atualização de dados em um documento
 - Os métodos `updateOne()` e `updateMany()` localizam o documento segundo os critérios especificados e fazem as alterações descritas
 - Diferença: quantidade de documentos afetada
 - Enquanto o `updateOne()` afeta somente um documento que atenda os critérios, o `updateMany()` afeta todos
 - O método `replaceOne()` localiza um único documento que atenda aos critérios especificados e o substitui por um novo documento.
 - O atributo `_id` do documento permanece o mesmo.

Criando um Banco de Dados MongoDB

- Coleções e Documentos (Update)

SQL

```
UPDATE alunos  
SET nome = "José", pais = "Brasil" ALTERAÇÃO  
WHERE id = 2; SELEÇÃO
```

MongoDB

```
db.alunos.updateOne({"_id": ObjectId("59111...")}, {"$set {"nome":  
"José", "pais": "Brasil"}})
```

```
db.alunos.updateMany({"pais": "Brasil"}, {"$unset {"pais": "Brasil"}})
```

```
db.alunos.replaceOne({"nome": "Maria"}, {"nome": "Raquel"})
```

Criando um Banco de Dados MongoDB

- Coleções e Documentos (Delete)

- O MongoDB possui dois métodos para a remoção de documentos.
- Os métodos `deleteOne()` e `deleteMany()` localizam o documento segundo os critérios especificados e o removem da base de dados.
 - Diferença: quantidade de documentos afetada.
 - Enquanto o `deleteOne()` afeta somente um documento que atenda os critérios, o `deleteMany()` afeta todos.

SQL

```
DELETE FROM alunos WHERE nome = "João"; SELEÇÃO
```

MongoDB

```
db.alunos.deleteOne({"nome": "José"})
```

```
db.alunos.deleteMany({})
```

Introdução ao MongoDB: PyMongo

- O PyMongo é uma API que contém ferramentas em linguagem python para operar bancos de dados MongoDB



- Possibilita que sejam desenvolvidas aplicações em linguagem python para manipular o banco de dados

Introdução ao MongoDB: PyMongo

- Prática

- Para instalar o PyMongo:

```
pip install pymongo
```

ou

```
python -m pip install pymongo
```

- Presume-se que o serviço do banco de dados já esteja rodando (*mongod*)
 - Material de referência para estudo:
 - <https://pymongo.readthedocs.io/en/stable/tutorial.html>
 - <https://www.w3schools.com/mongodb>



Introdução ao MongoDB: PyMongo

- Prática

- Google Colab (sugestão de uso)

```
# install and start server
!apt install mongodb >log
!service mongodb start
```

```
from pymongo import MongoClient
client = MongoClient()
```



Introdução ao MongoDB: PyMongo

- Tarefa:

- Desenvolver uma aplicação em python, recorrendo à API PyMongo, que seja capaz de ler o arquivo de entrada abaixo, o converta em padrão mongodb, criando um banco de dados e inserindo os dados no mesmo. Pretende-se que os dados possam ser consultados (“query”) a partir do banco de dados criado.

- Entrada: Lista de discentes ingressantes na UFRN em 2022

<https://dados.ufrn.br/dataset/discentes/resource/14afbb6c-395e-411c-b24d-0e494cb95866>

- As colunas estão ordenadas da seguinte maneira:

```
[ "matricula"; "ano_ingresso"; "periodo_ingresso"; "id_curso"; "id_unidade"; "id_unidade_gestora"; "nome_discente"; "sexo"; "forma_ingresso"; "tipo_discente"; "status"; "sigla_nivel_ensino"; "nivel_ensino"; "nome_curso"; "modalidade_educacao"; "nome_unidade"; "nome_unidade_gestora" ]
```

- Exemplos de consultas (queries):

1. Listar todos os alunos que ingressaram por meio do SiSU
2. Listar todos os alunos que ingressaram no mestrado
3. Listar todos os alunos de uma unidade gestora
4. Listar todos os alunos regulares
5. Listar todos os alunos pelo status (cancelado, ativo, etc.)