



IFPR - LONDRINA

BREVÍSSIMA INTRODUÇÃO AO MONGODB

Diogo Olsen.



mongoDB®

Agenda



INTRODUÇÃO

Apresentação do MongoDB, sua filosofia e classificação.



USO

Principais comandos CRUD - Create, Read, Update e Delete.



COMPASS - ALTAS

Infraestrutura oferecida pelo MongoDB em nuvem e local.



REFERÊNCIAS / INDICAÇÃO

Referências da apresentação.




















MONGODB

INTRODUÇÃO

- Origem do nome MongoDB: derivado da palavra *humongous* - enorme/gigantesco em inglês.
- Trata-se de um software de Banco de Dados, de código aberto, multiplataforma mantido pela Mongo Inc - <https://www.mongodb.com/>
- Classificado como NoSql (*Not Only SQL* - entendido como não relacional ou não SQL).
 - O BD NoSQL mais usado atualmente - 5º na classificação geral.

MONGODB

INTRODUÇÃO

Rank			DBMS	Database Model	Score		
Apr 2021	Mar 2021	Apr 2020			Apr 2021	Mar 2021	Apr 2020
1.	1.	1.	Oracle 	Relational, Multi-model 	1274.92	-46.82	-70.51
2.	2.	2.	MySQL 	Relational, Multi-model 	1220.69	-34.14	-47.66
3.	3.	3.	Microsoft SQL Server 	Relational, Multi-model 	1007.97	-7.33	-75.46
4.	4.	4.	PostgreSQL 	Relational, Multi-model 	553.52	+4.23	+43.66
5.	5.	5.	MongoDB 	Document, Multi-model 	469.97	+7.58	+31.54
6.	6.	6.	IBM Db2 	Relational, Multi-model 	157.78	+1.77	-7.85
7.	7.	 8.	Redis 	Key-value, Multi-model 	155.89	+1.74	+11.08
8.	8.	 7.	Elasticsearch 	Search engine, Multi-model 	152.18	-0.16	+3.27
9.	9.	9.	SQLite 	Relational	125.06	+2.42	+2.87
10.	10.	10.	Microsoft Access	Relational	116.72	-1.41	-5.19

- Fonte: <https://db-engines.com/en/ranking> - 06/04/2021

MONGODB

INTRODUÇÃO

- O princípio filosófico mais importante na base do MongoDB é a noção de que:
 - *um só tamanho não é ideal para todos.*
- Durante muito tempo os RDBMS (ou Sistema de Gerenciamento de Banco de Dados Relacional) eram a forma hegemônica de armazenar dados em sistemas computacionais, pouco importando se os dados fossem ou não adequados ao modelo relacional.

MONGODB

INTRODUÇÃO

- Isso ocorria pois é mais seguro armazenar dados, mesmo que organizados de "qualquer" maneira, em um RDBMS do que em arquivos/disco, além da facilidade da busca de dados.
- Porém, pode ser complexo armazenar dados heterogêneos e, principalmente, dados que mudem com frequência em RDBMS - SQL.

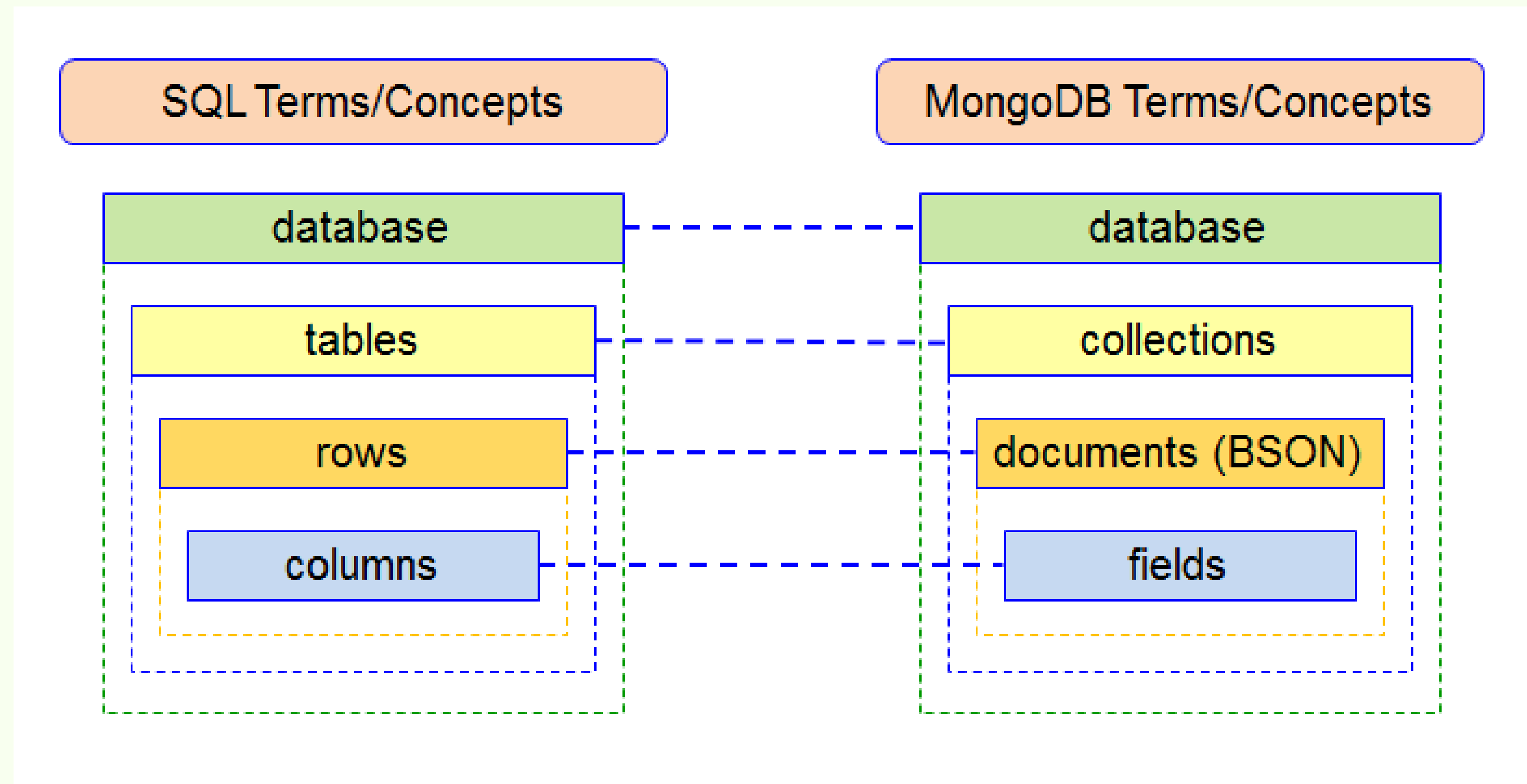
MONGODB

INTRODUÇÃO

- Assim, o MongoDB foi projetado no modelo NoSQL baseado em documentos.
 - Trabalha com coleções de documentos e documentos embutidos (*embedded documents*) em vez de tabelas com colunas e linhas

MONGODB

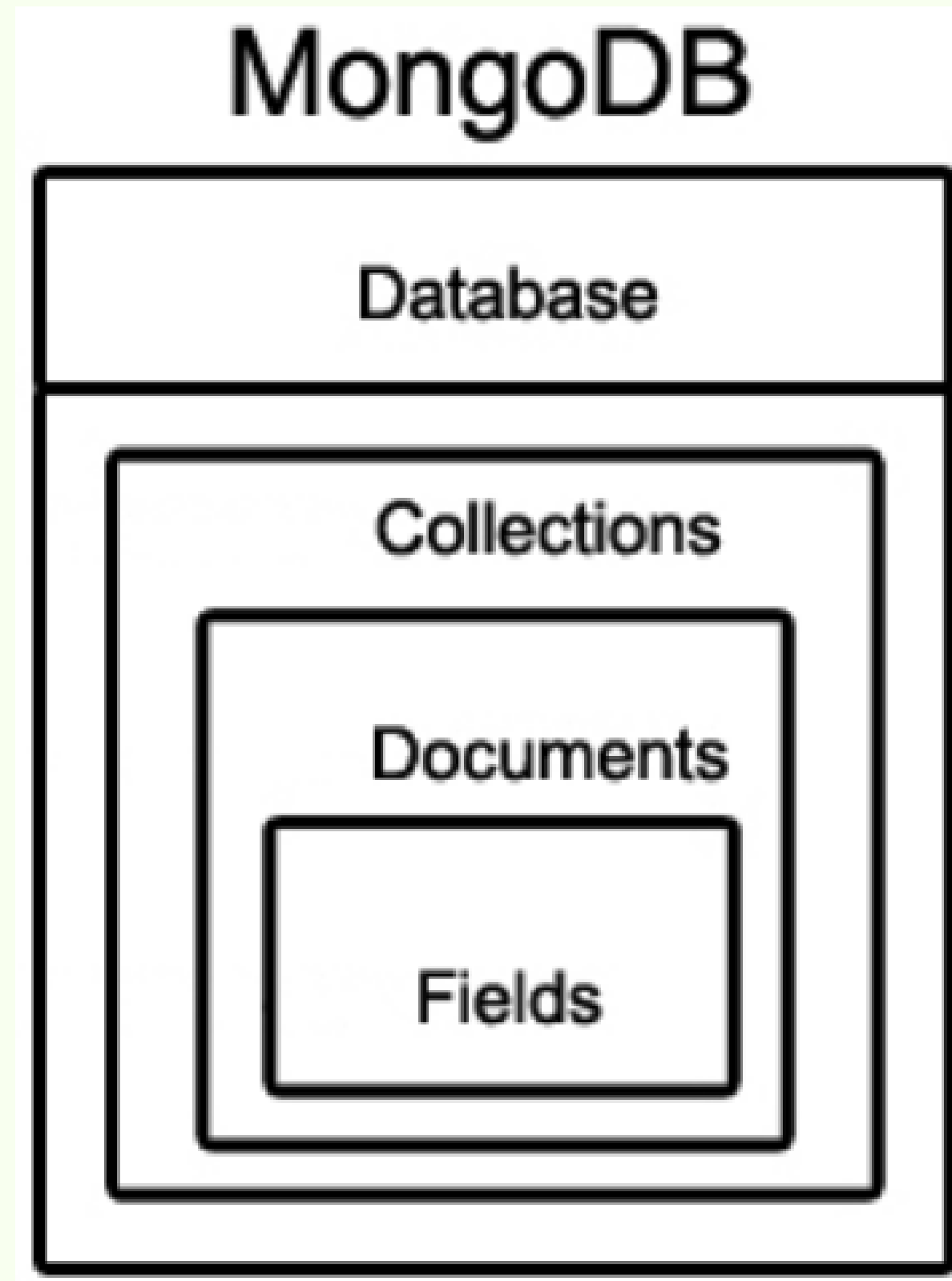
INTRODUÇÃO



- Fonte: <http://sql-vs-nosql.blogspot.com/2013/11/indexes-comparison-mongodb-vs-mssqlserver.html>

MONGODB

INTRODUÇÃO



- Fonte: datacamp - Introduction to MongoDB in Python

MONGODB

INTRODUÇÃO

MongoDB	JSON	Python
Databases	Objects	Dictionaries
↳ Collections	Arrays	Lists
↳ ↳ Documents	Objects	Dictionaries
↳ ↳ ↳ Subdocuments	Objects	Dictionaries
↳ ↳ ↳ Values	Value types	Value types + datetime , regex...

- Fonte: datacamp - Introduction to MongoDB in Python

MONGODB

INTRODUÇÃO

```
_id: ObjectId("6009805040dac34747d94927")
name: "Abóbora Paulista"
searchable: "ABOBORA PAULISTA"
type: "Vegetal"
unity: "g"
✓ factorsLog: Array
  ✓ 0: Object
    date: 2021-01-21T10:23:28.595+00:00
    cookingFactor: 0.8
    safetyMargin: 1.03
    actualFactor: 0.824000000000000001
✓ costLog: Array
  > 0: Object
    establishedCostPer1K: 2
```

MONGODB

INTRODUÇÃO

- Assim, o MongoDB foi projetado no modelo NoSQL baseado em documentos.
 - Trabalha com coleções de documentos e documentos embutidos (*embedded documents*) em vez de tabelas com colunas e linhas
 - Baseado em JSON (Java Script Object Notation) - BSON (Binary JSON).

MONGODB

INTRODUÇÃO

```
{
  "_id": {
    "$oid": "6009805040dac34747d94927"
  },
  "name": "Abóbora Paulista",
  "searchable": "ABOBORA PAULISTA",
  "type": "Vegetal",
  "unity": "g",
  "factorsLog": [{
    "date": {"$date": "2021-01-01T00:00:00Z"},
    "cookingFactor": 0.8,
    "safetyMargin": 1.03,
    "actualFactor": 0.8240000000000001
  }],
  "costLog": [{"date": "2021-01-01T00:00:00Z", "cost": 1.0}],
  "establishedCostPer1K": 2
}
```

MONGODB

INTRODUÇÃO

- Documento embutido

```
{  
  "_id": {  
    "$oid": "6009805040dac34747d94927"  
  },  
  "name": "Abóbora Paulista",  
  "searchable": "ABOBORA PAULISTA",  
  "type": "Vegetal",  
  ...  
}
```

MONGODB

INTRODUÇÃO

- Baseado em Chave-Valor

```
{
```

```
...
```

```
  "name": "Abóbora Paulista",
```

```
  "searchable": "ABOBORA PAULISTA",
```

```
  "type": "Vegetal",
```

```
  "unity": "g",
```

```
...
```


MONGODB

INTRODUÇÃO

- Possibilita conjunto de documentos embutidos

...

```
"factorsLog": [{  
  "date": {  
    "$date": "2021-01-21T10:23:28.595Z"  
  },  
  ...  
}]
```

...

MONGODB

INTRODUÇÃO

- Além disso, no MongoDB, a ausência de suporte inato a atomicidade e transações diminui a necessidade de Locks, diminuindo a complexidade e melhorando o desempenho.
- MongoDB tende a ser mais **simples**, **rápido** e **escalável** (principalmente horizontalmente) do que RDBMS.
- Possibilita a consulta de documentos através de métodos avançados de agrupamento e filtragem (MapReduce) e também permite redundância e inconsistência.

MONGODB

INTRODUÇÃO

- Escalabilidade horizontal é a possibilidade de usar mais máquinas menores em conjunto (clusters) em vez de usar uma máquina mais potente (escalabilidade vertical).
- Obs: MYSQL - por padrão - também não oferece suporte a transações.

MONGODB

INTRODUÇÃO

- MongoDB não é a solução ideal para tudo! Nem o SQL!
- Soluções híbridas podem resolver diversas soluções de maneira mais elegante e eficiente.
- Sistemas financeiros, como de cartões de crédito, podem exigir sistemas relacionais de BD!
- A Bolsa de NY usa um sistema de transações baseada em documentos estruturados para operar - RDBMS não possuem a velocidade necessária - **sem referência.**

MONGODB

COMPASS

The screenshot displays the MongoDB Compass application window. The title bar reads "MongoDB Compass - 127.0.0.1:27017/CozinhaPetDB.Ingredients". The interface is divided into a left sidebar and a main content area.

Left Sidebar:

- Local:** Shows connection details: HOST (127.0.0.1:27017), CLUSTER (Standalone), and EDITION (MongoDB 3.6.8 Community).
- Database Explorer:** Lists databases and collections. Under "CozinhaPetDB", the "Ingredients" collection is selected.

Main Content Area:

- Collection Tab:** "CozinhaPetDB.Ingredients Documents".
- Summary:** DOCUMENTS 22, TOTAL SIZE 7.6KB, AVG. SIZE 352B. INDEXES 1, TOTAL SIZE 36.0KB, AVG. SIZE 36.0KB.
- Navigation:** Tabs for Documents, Aggregations, Schema, Explain Plan, Indexes, and Validation.
- Actions:** FILTER, OPTIONS, FIND, RESET, and a menu icon.
- Data View:** Shows a list of documents. The first document is expanded, showing its JSON structure:

```
{
  "_id": ObjectId("6009805040dac34747d94927"),
  "name": "Abóbora Paulista",
  "searchable": "ABOBORA PAULISTA",
  "type": "Vegetal",
  "unity": "g",
  "factorsLog": Array,
  "costLog": Array,
  "establishedCos...": 2
}
```

The second document is also visible below it.

MONGODB

COMPASS

- Bases de dados.
- Coleções de documentos.
- Documentos.
- Documentos embutidos.
- Pares Chave-Valor.

MONGODB

ATLAS

- Solução em nuvem, baseada no MongoDB, compatível com AWS, Google Cloud e Azure.
- Versão gratuita para testes até 5GB - Suficiente para este trabalho.

MONGODB

USO - INSTALAÇÃO

- <https://www.mongodb.com/try/download/community>

MONGODB

USO

- **\$ mongo** - Inicializa o cliente MongoDB
- **> show databases** - Exibe as bases de dados;
- **> use nobel** - Seleciona\Cria a base de dados 'nobel';
- **> show collections** - Exibe as coleções de documentos na base de dados;
- **> db.prizes.find()** - Exibe todos os documentos na coleção 'prizes'

MONGODB

USO

- > **use AulaCAOO** - Cria a BD 'AulaCAOO';
- > **db.turma.insert({'tipo':'Professor', 'nome': 'Diogo'})** - Cria a coleção 'turma' e insere o primeiro documento;
- > **db.turma.insert([{'tipo':'Estudante', 'nome': 'Ana'}, {'tipo':'Estudante', 'nome': 'Alex'}])** - insere um array de documentos;

MONGODB

USO

- **db.turma.find({'tipo':'Estudante'})** - Busca documentos onde o 'tipo' seja 'Estudante'

```
{ "_id" : ObjectId("606c37c1d0bb629d4c4a2856"), "tipo" : "Estudante", "nome" : "Ana" }  
{ "_id" : ObjectId("606c37c1d0bb629d4c4a2857"), "tipo" : "Estudante", "nome" : "Alex" }
```
- **db.turma.findOne({'tipo':'Estudante'})** - Busca um doc. cujo 'tipo' seja 'Estudante'

```
{  
  "_id" : ObjectId("606c37c1d0bb629d4c4a2856"),  
  "tipo" : "Estudante",  
  "nome" : "Ana"  
}
```

MONGODB

USO

- > **db.turma.updateOne({'nome':'Diogo'}, { \$set: {'nome': 'Diogo Olsen'}})** - Atualiza um documento;
- > **db.turma.update({'tipo':'Estudante'}, { \$set: {'tipo': 'Aluno'}}, {multi:true})** - Atualiza todos os documentos encontrados;

MONGODB

USO

- > **db.turma.remove({'nome': 'Diogo Olsen'})** - Remove um documento;
- > **db.turma.remove({'tipo': 'Aluno'})** - Remove todos os alunos;
- > **db.turma.find()** - Nenhum documento;

MONGODB

USO

- > **db.turma.drop()** - Remove a coleção 'turma'
- > **db.dropDatabase()** - Remove a base de dados AulaCAOO
{ "dropped" : "AulaCAOO", "ok" : 1 }

MONGODB

REFERÊNCIAS

- Introdução ao MongoDB - David Hows, Peter Membrey e Eelco Plugge
- MongoDB para Iniciantes - Luiz F. Duarte Jr.
- Introduction to MongoDB in Python - Datacamp
- <https://docs.mongodb.com/>

MONGODB

INDICAÇÃO DE LEITURA

[https://www.mongodb.com/blog/post/building-with-patterns-a-summary?
utm_campaign=Int_EM_ONB_FT1a_10_19_WW&utm_source=Eloqua&utm_medium=email
&utm_term=Designing%20a%20MongoDB%20schema%20for%20Atlas](https://www.mongodb.com/blog/post/building-with-patterns-a-summary?utm_campaign=Int_EM_ONB_FT1a_10_19_WW&utm_source=Eloqua&utm_medium=email&utm_term=Designing%20a%20MongoDB%20schema%20for%20Atlas)

MONGODB

OBRIGADO!
