

A decorative graphic on the left side of the slide consisting of two overlapping parallelograms. The front one is blue and the back one is a light green. They are positioned diagonally, with the blue one partially covering the green one.

# MongoDB

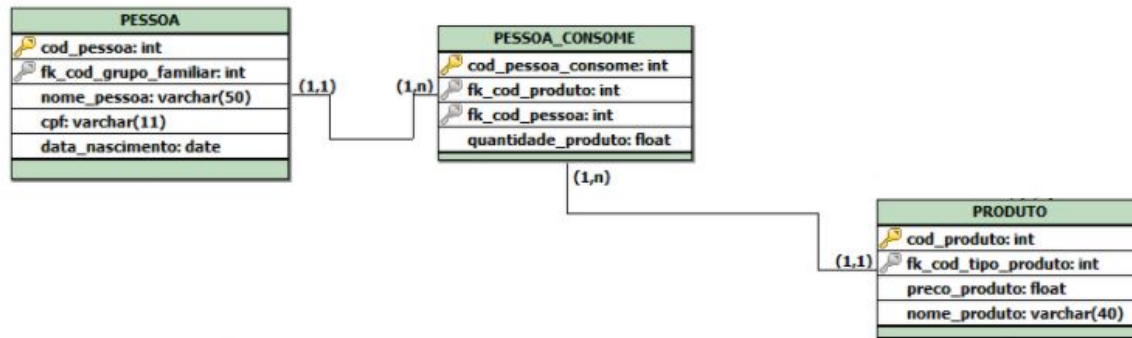
Kelvin Lehrback



# MongoDB - Características Gerais

- NoSQL
- Código fonte aberto
- Orientado a documentos
- Não possui “inner join”
- A estrutura das collections (podemos fazer uma comparação com as tabelas em SQL) é facilmente modificável, facilitando a alteração de sua estrutura
- Integração com SQL através do módulo de BI

# Modelo lógico vs Modelo MongoDB



```
_id: 1
nome_pessoa: "Valdeci G"
cpf: "61576870880"
data_nascimento: "1977-09-03"
~ produtos_consumidos: Array
  ~ 0: Object
    nome_produto: "Arroz"
    preco_produto: 7.79
    quantidade_produto: 100
  ~ 1: Object
    nome_produto: "Peixe"
    preco_produto: 37.9
    quantidade_produto: 250
```

```
_id: 6
nome_pessoa: "Merlin de Avalon"
cpf: "93397782007"
data_nascimento: "1960-05-17"
~ produtos_consumidos: Array
  ~ 0: Object
    nome_produto: "Peixe"
    preco_produto: 37.9
    quantidade_produto: 300
  ~ 1: Object
    nome_produto: "Feijão"
    preco_produto: 5.99
    quantidade_produto: 250
```



# Sintaxe de conexão

## Importando biblioteca e realizando configurações para conexão com o MongoDB

```
[1] import pymongo
```

```
[2] client = pymongo.MongoClient("mongodb+srv://Kelvin:mJzr29PLVbhgfmt5@cluster0.tdkfu.mongodb.net/<dbname>?retryWrites=true&w=majority")
```

```
[3] db = client.test
```

## Acessando o banco de dados desejado

```
[4] db = client.get_database('provisoes_de_emergencia')
```

# Sintaxe de inclusão

```
[ ] objeto_produto = {  
    "nome_produto": "Lasanha",  
    "preco_produto": 10.90  
}  
db.produto.insert_one(objeto_produto)
```

```
produtos = [  
    {  
        "_id": 1,  
        "nome_produto": "Arroz",  
        "preco_produto": 7.79  
    },  
    {  
        "_id": 2,  
        "nome_produto": "Feijão",  
        "preco_produto": 5.99  
    },  
    {  
        "_id": 3,  
        "nome_produto": "Peixe",  
        "preco_produto": 37.90  
    }  
]
```

```
        "preco_produto": 8.59  
    },  
    {  
        "_id": 10,  
        "nome_produto": "Batata Monalise",  
        "preco_produto": 3.85  
    }  
]  
db.produto.insert_many(produtos)
```

# Sintaxe de seleção

```
list(db.pessoa.find({}))
```

```
[{'_id': ObjectId('5f74d75f61f079630418a4b3'),  
  'cpf': '61576870880',  
  'data_nascimento': '1977-09-03',  
  'nome_pessoa': 'Valdeci G'},  
 {'_id': ObjectId('5f74d75f61f079630418a4b4'),  
  'cpf': '80663532787',  
  'data_nascimento': '2007-12-20',  
  'nome_pessoa': 'Clara L'},  
 {'_id': ObjectId('5f74d75f61f079630418a4b5'),  
  'cpf': '6410440979',  
  'data_nascimento': '1979-07-05',  
  'nome_pessoa': 'Kelen L'},  
 {'_id': ObjectId('5f74d75f61f079630418a4b6'),  
  'cpf': '69505791607',  
  'data_nascimento': '1986-12-02',  
  'nome_pessoa': 'Sebastião da Silva'},  
 {'_id': ObjectId('5f74d75f61f079630418a4b7'),  
  'cpf': '14101087830',
```

```
list(db.pessoa_consoma.find({}))
```

```
[{'_id': 1,  
  'cpf': '61576870880',  
  'data_nascimento': '1977-09-03',  
  'nome_pessoa': 'Valdeci G',  
  'produtos_consumidos': [{'nome_produto': 'Arroz',  
    'preco_produto': 7.79,  
    'quantidade_produto': 100},  
    {'nome_produto': 'Peixe',  
    'preco_produto': 37.9,  
    'quantidade_produto': 250}]},  
 {'_id': 2,  
  'cpf': '80663532787',  
  'data_nascimento': '2007-12-20',  
  'nome_pessoa': 'Clara L',  
  'produtos_consumidos': [{'nome_produto': 'Feijão',  
    'preco_produto': 5.99,  
    'quantidade_produto': 150},  
    {'nome_produto': 'Farinha', 'preco_produto': 5.39}]},  
 {'_id': 3,  
  'cpf': '6410440979',  
  'data_nascimento': '1979-07-05',
```

## Buscando um documento em específico (utilizando filtros)

```
db.produto.find_one({'nome_produto': 'Arroz'})
```

```
{'_id': 1, 'nome_produto': 'Arroz', 'preco_produto': 7.79}
```

```
list(db.produto.find())
```

```
[{'_id': 1, 'nome_produto': 'Arroz', 'preco_produto': 7.79},  
 {'_id': 2, 'nome_produto': 'Feijão', 'preco_produto': 5.99},  
 {'_id': 3, 'nome_produto': 'Peixe', 'preco_produto': 37.9},  
 {'_id': 4, 'nome_produto': 'Farinha', 'preco_produto': 5.39},  
 {'_id': 5, 'nome_produto': 'Cloro', 'preco_produto': 10.79},  
 {'_id': 6, 'nome_produto': 'Sabão em pó', 'preco_produto': 7.9},  
 {'_id': 7, 'nome_produto': 'Tomate Carmem', 'preco_produto': 3.59},  
 {'_id': 8, 'nome_produto': 'Dorflex', 'preco_produto': 3.68},  
 {'_id': 9, 'nome_produto': 'Vitamina C', 'preco_produto': 8.59},  
 {'_id': 10, 'nome_produto': 'Batata Monalisa', 'preco_produto': 3.85}]
```



# Sintaxe de exclusão

## Deletando documentos

```
[ ] db.produto.delete_one({'preco_produto': 'Dorflex'})
```

```
[ ] db.produto.delete_many({'preco_produto': {'$gte': 8.59}})
```



# Sintaxe de atualização

## Atualizando documentos

```
▶ update_produto = {  
    'preço': 9.99  
}  
db.produto.update_one({'nome_produto': 'Arroz Sepé'}, {'$set': update_produto})
```

```
[ ] update_pessoa = {  
    'nome_pessoa': 'Sebastião da Silva'  
}  
db.pessoa.update_one({'cpf': '69505791607'}, {'$set': update_pessoa})
```





# Pontos positivos

- Liberdade de escolha do sistema operacional
- Estrutura flexível, facilitando o “update” dos dados e coleções
- Uma curva de aprendizado descomplicada
- Ideal para startups
- Possui um servidor em nuvem gratuito (ideal para estudantes), pagando apenas por benefícios extras e suporte



# Pontos negativos

- Informações espalhadas, ou seja, falta de joins.
- Falta de amplo suporte
- Alto consumo de RAM
- Seu principal ponto pode ser sua principal fraqueza: Documentos complexos atrapalham!



# Referências

## Documentação oficial:

- <https://docs.mongodb.com/guides/>

## Blogs:

- <http://lucasmaiaesilva.com.br/posts/usando-o-mongo-shell/>
- <https://www.itexto.com.br/devkico/?p=706>
- <https://www.youtube.com/watch?v=jC7imm4ZLRY&t>
- <https://db-engines.com/en/system/MongoDB>

## Vídeos:

- [https://www.youtube.com/watch?v=VQnmcBnguPY&ab\\_channel=IndianPythonista](https://www.youtube.com/watch?v=VQnmcBnguPY&ab_channel=IndianPythonista)
- <https://www.youtube.com/watch?v=jC7imm4ZLRY&t>