

Introdução ao MongoDB com Python



Geraldo!

Desenvolvedor na InteSoftware

Vegetariano, diabético, entusiasta, desenhista, músico,
compositor e exagerado!

About me!

- Estudante de CienCom
- Coordenador do Grupy-RN
- Programador desde 2012
- Usuário Linux e Android
- Não uso WhatsApp!!!1!
- Membro ativo nas comunidades JerimumHS, GIM e Python (sempre!)



/geraldoc.py



/victormatheusc



@exaGeraldo



/victormatheusc

1. Introdução

SQL? NoSQL? SOS!

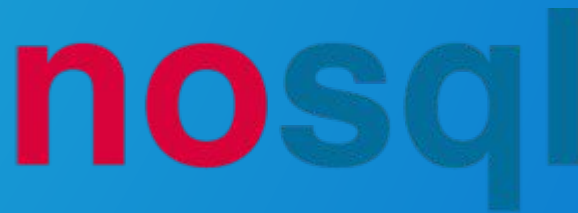
SQL

"Structured Query Language" (Linguagem de Consulta Estruturada). É uma linguagem de consulta a banco de dados relacionais. Geralmente, bancos SQL seguem uma modelagem relacional.



NoSQL

“Not Only SQL” é o termo utilizado para banco de dados não relacionais de alto desempenho, onde geralmente não é utilizado o SQL como linguagem de consulta. Foi criado para ter uma performance melhor e uma escalabilidade mais horizontal para suprir necessidades onde os bancos relacionais não são eficazes

The logo for NoSQL, with 'nosql' in a bold, lowercase, sans-serif font. The 'no' is red and the 'sql' is dark blue.

Tipos de bancos NoSQL

Documento

Os dados são armazenados como documentos no formato chave-valor, com o padrão JSON.

Colunas

Os dados são armazenados em linhas particulares de tabela no disco. Pode suportar várias linhas e colunas, e também sub-colunas.

Tipos de bancos NoSQL

Grafos

Os dados são armazenados na forma de grafos (vértices e arestas).

Chave-Valor

Os dados são armazenados no formato chave-valor. Um determinado valor só poderá ser acessado através de uma chave única.

Comparando

SQL

- Armazenamento tabular
- SQL como linguagem padrão
- Normalização e desnormalização
- Esquema rígido



NoSQL

- Schemaless
- Nível de consistência variado
- Suporta SQL e linguagens próprias
- Diferente formas de armazenamento

Não existe bala de prata!

Por ser considerado ainda novo, o NoSQL é encarado com cautela pelos tradicionais do SQL. Uma comparação é sempre importante para avaliar qual tecnologia se adequa melhor para sua ideia!

2. MongoDB

O que muléstia é Mongo e pra que serve?



12

Programando!

```
from pymongo import MongoClient
import datetime

documento = {
    "nome": "Victor Matheus",
    "telegram": "@exageraldo",
    "grupos": ["grupy-rn", "jerimumhs"],
    "data_cadastro": datetime.datetime.now()
}

cliente = MongoClient('localhost', 27017)
banco = cliente['banco']
colecacao = banco['colecacao']

# Inserir
colecacao.insert_one(documento)

# Buscar
colecacao.find_one({"nome": "Victor Matheus"})

# Atualizar
colecacao.update_one({'_id': 1}, {'$set': {'telegram': '@exaGeraldo'}})

# Deletar
colecacao.delete_one({"_id": 1})
```

Programando!

```
from pymongo import MongoClient

class NomeDaClasse(MongoClient):
    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)
        self.banco = self.banco
        self.colecao = self.banco.colecao

    def inserir(self, documento):
        self.colecao.insert_one(documento)

    def buscar(self, nome):
        return self.colecao.find_one({"nome": nome})

    def atualizar(self, nome_velho, nome_novo):
        self.colecao.update_one(
            {'nome': nome_velho}, {'$set': {'nome': 'nome_novo'}})

    def deletar(self, nome):
        self.colecao.delete_one({"nome": nome})

item = NomeDaClasse('localhost', 27017)
novo_item = {
    "nome": "Victor Matheus",
    "telegram": "@exageraldo",
    "grupos": ["grupy-rn", "jerimumhs"],
    "data_cadastro": datetime.datetime.now()
}

item.inserir(novo_item)
item.buscar("Victor Matheus")
item.atualizar("Victor Matheus", "Geraldo")
item.deletar("Geraldo")
```



Obrigado!

Alguma dúvida?

Entre em contato comigo pelo e-mail
victormatheuscastro@gmail.com ou no Telegram
[@exaGeraldo](#)