

Documentação do Projeto CRUD MongoDB

Este documento descreve o funcionamento do programa em Python que implementa as operações CRUD (Create, Read, Update, Delete) em um banco de dados MongoDB. O projeto realiza a inserção, leitura, atualização e remoção de documentos na coleção 'pedidos'. Além disso, a documentação inclui instruções detalhadas para rodar o projeto localmente e informações sobre como configurar o ambiente MongoDB. Abaixo estão as instruções completas para executar o projeto e compreender o código-fonte, bem como a explicação de cada parte do projeto.

1. Instalar dependências
2. Configurar o banco de dados MongoDB
3. Rodar o código Python
4. Funções CRUD implementadas

Como Rodar o Projeto

1. Clone o repositório:

```
git clone https://github.com/enka063/atividade-python-mysql
```

2. Instale as dependências do Python:

```
pip install pymongo
```

3. Instale o MongoDB (caso não tenha)

4. Inicie o MongoDB: mongod

5. Execute o script Python:

```
python main.py
```

Funções CRUD Implementadas

As seguintes funções foram implementadas no código para interagir com o banco de dados MongoDB:

1. ****Criar pedido**** (create_mongo):

Esta função insere um novo pedido na coleção 'pedidos'. O pedido contém informações como ID, nome do cliente, itens do pedido e valor total.

2. ****Ler pedidos**** (read_mongo):

Esta função retorna todos os pedidos na coleção 'pedidos'. Ela faz uma busca para listar todos os documentos presentes no banco.

3. ****Atualizar pedido**** (update_mongo):

A função de atualização modifica o valor total de um pedido, identificando o pedido pelo seu ID e alterando o campo 'valor_total'.

4. ****Deletar pedido**** (delete_mongo):

Esta função remove um pedido da coleção 'pedidos'. O pedido é deletado com base no ID fornecido. Abaixo estão exemplos de como cada operação é utilizada no código.

Exemplo de Uso das Funções

- ****Criar pedido****:

A função 'create_mongo()' insere um novo pedido na coleção 'pedidos'. Exemplo de dados inseridos:

```
{  
  "id": "1",  
  "cliente": "João Silva",  
  "itens": [  
    {"nome": "Pizza", "quantidade": 1, "preco": 30.00},  
    {"nome": "Refrigerante", "quantidade": 2, "preco": 5.00}  
  ],  
  "valor_total": 40.00  
}
```

- ****Ler pedidos****:

A função 'read_mongo()' retorna todos os pedidos no banco. Exemplo de saída:

```
[  
  {  
    "id": "1",  
    "cliente": "João Silva",
```

```
"itens": [  
    {"nome": "Pizza", "quantidade": 1, "preco": 30.00},  
    {"nome": "Refrigerante", "quantidade": 2, "preco": 5.00}  
],  
"valor_total": 40.00  
},  
...  
]
```

- **Atualizar pedido**:

A função 'update_mongo()' atualiza o valor total de um pedido. Exemplo:

```
db.pedidos.update_one({"id": "1"}, {"$set": {"valor_total": 50.00}})
```

- **Deletar pedido**:

A função 'delete_mongo()' remove um pedido com base no ID. Exemplo:

```
db.pedidos.delete_one({"id": "1"})
```

Testes que foram realizados

Os testes realizados verificam o funcionamento das funções CRUD no banco de dados MongoDB. Para o sr. testar o programa, execute o código e verifique as saídas dos pedidos criados, atualizados, lidos e deletados. Cada uma das funções foram testadas com dados válidos para garantir que as operações de inserção, leitura, atualização e remoção estejam funcionando corretamente. No MongoDB, as coleções são criadas

automaticamente quando um documento é inserido pela primeira vez, e o código já está configurado para lidar com isso de forma transparente.

Conclusão

Essa atividade demonstrou a implementação de um CRUD funcional utilizando o banco de dados “MongoDB” com Python. Esse programa possibilita gerenciar pedidos de um sistema de vendas de forma simples e eficiente. As funções implementadas oferecem a capacidade de realizar todas as operações essenciais para a manipulação de dados no MongoDB. O código-fonte, juntamente com a documentação, estará disponível na plataforma do GitHub para a utilização e modificação conforme for necessário.