

Atividade 3: Banco de dados NoSQL

Atividade: Usar FastAPI para implementar uma busca usando um índice criado com `createIndex()`

1. Preparação do Ambiente:

FastAPI, Uvicorn para rodar a aplicação, e Pymongo para a conexão com o MongoDB.

Terminal

`pip install "fastapi[all]" uvicorn pymongo`

```
PS C:\Users\tully> pip install fastapi uvicorn pymongo
Collecting fastapi
  Downloading fastapi-0.116.1-py3-none-any.whl.metadata (28 kB)
Collecting uvicorn
  Downloading uvicorn-0.35.0-py3-none-any.whl.metadata (6.5 kB)
Requirement already satisfied: pymongo in c:\users\tully\appdata\local\programs\python\python313\lib\site-packages (4.13.2)
Collecting starlette<0.48.0,>=0.40.0 (from fastapi)
  Downloading starlette-0.47.2-py3-none-any.whl.metadata (6.2 kB)
Collecting pydantic!=1.8,!>=1.8.1,!>=2.0.0,!>=2.0.1,!>=2.1.0,<3.0.0,>=1.7.4 (from fastapi)
  Downloading pydantic-2.11.7-py3-none-any.whl.metadata (67 kB)
Collecting typing-extensions>=4.8.0 (from fastapi)
  Downloading typing_extensions-4.14.1-py3-none-any.whl.metadata (3.0 kB)
Collecting click>=7.0 (from uvicorn)
  Downloading click-8.2.1-py3-none-any.whl.metadata (2.5 kB)
Collecting h11>=0.8 (from uvicorn)
  Downloading h11-0.16.0-py3-none-any.whl.metadata (8.3 kB)
Requirement already satisfied: dnspython<3.0.0,>=1.16.0 in c:\users\tully\appdata\local\programs\python\python313\lib\site-packages (from click>=7.0->uvicorn)
Collecting colorama (from click>=7.0->uvicorn)
  Downloading colorama-0.4.6-py2.py3-none-any.whl.metadata (17 kB)
Collecting annotated-types>=0.6.0 (from pydantic!=1.8,!>=1.8.1,!>=2.0.0,!>=2.0.1,!>=2.1.0,<3.0.0,>=1.7.4->fastapi)
  Downloading annotated_types-0.7.0-py3-none-any.whl.metadata (15 kB)
Collecting pydantic_core==2.33.2 (from pydantic!=1.8,!>=1.8.1,!>=2.0.0,!>=2.0.1,!>=2.1.0,<3.0.0,>=1.7.4->fastapi)
  Downloading pydantic_core-2.33.2-cp313-cp313-win_amd64.whl.metadata (6.9 kB)
Collecting typing-inspection>=0.4.0 (from pydantic!=1.8,!>=1.8.1,!>=2.0.0,!>=2.0.1,!>=2.1.0,<3.0.0,>=1.7.4->fastapi)
  Downloading typing_inspection-0.4.1-py3-none-any.whl.metadata (2.6 kB)
Collecting anyio<5,>=3.6.2 (from starlette<0.48.0,>=0.40.0->fastapi)
  Downloading anyio-4.10.0-py3-none-any.whl.metadata (4.0 kB)
Collecting idna>=2.8 (from anyio<5,>=3.6.2->starlette<0.48.0,>=0.40.0->fastapi)
  Downloading idna-3.10-py3-none-any.whl.metadata (10 kB)
Collecting sniffio>=1.1 (from anyio<5,>=3.6.2->starlette<0.48.0,>=0.40.0->fastapi)
  Downloading sniffio-1.3.1-py3-none-any.whl.metadata (3.9 kB)
Downloading fastapi-0.116.1-py3-none-any.whl (95 kB)
Downloading uvicorn-0.35.0-py3-none-any.whl (66 kB)
Downloading click-8.2.1-py3-none-any.whl (102 kB)
Downloading h11-0.16.0-py3-none-any.whl (37 kB)
Downloading pydantic-2.11.7-py3-none-any.whl (444 kB)
Downloading pydantic_core-2.33.2-cp313-cp313-win_amd64.whl (2.0 MB)
2.0/2.0 MB 11.1 MB/s eta 0:00:00
Downloading starlette-0.47.2-py3-none-any.whl (72 kB)
Downloading typing_extensions-4.14.1-py3-none-any.whl (43 kB)
Downloading annotated_types-0.7.0-py3-none-any.whl (13 kB)
```

2. Texto de rascunho: `pip install -r rascunho.txt`

3. Código da API para implementar: `teste1.py`

```

C:\Users\tully> from.py > Cliente > Config
1  from pydantic import BaseModel, Field, validator
2  from typing import Optional
3  from datetime import date
4
5  class UltimaCompra(BaseModel):
6      produto: str = Field(..., example="Batom MAC")
7      valor: float = Field(..., gt=0, example=189.9)
8      data: str = Field(..., regex="^\d{4}-\d{2}-\d{2}$", example="2023-11-15")
9
10     @validator('data')
11     def validar_data(cls, v):
12         try:
13             return date.fromisoformat(v)
14         except ValueError:
15             raise ValueError("Formato de data inválido. Use YYYY-MM-DD")
16
17     class ClienteBase(BaseModel):
18         id: str = Field(..., min_length=5, max_length=10, example="C1001")
19         nome: str = Field(..., min_length=3, example="Ana Silva")
20         idade: int = Field(..., gt=0, lt=120, example=34)
21         ultima_compra: UltimaCompra
22
23     class ClienteCreate(ClienteBase):
24         pass
25
26     class ClienteUpdate(BaseModel):
27         nome: Optional[str] = Field(None, min_length=3, example="Ana Silva Santos")
28         idade: Optional[int] = Field(None, gt=0, lt=120, example=35)
29         ultima_compra: Optional[UltimaCompra] = None
30
31     class Cliente(ClienteBase):
32         class Config:
33             schema_extra = {
34                 "example": {
35                     "id": "C1001",
36                     "nome": "Ana Silva",
37                     "idade": 34,
38                     "ultima_compra": {
39                         "produto": "Batom MAC",
40                         "valor": 189.9,
41                         "data": "2023-11-15"
42                     }
43                 }
44             }
45
46

```

3. Execute a API:

vicorn teste1:app --reload

```

PS C:\Users\tully> uvicorn teste1:app --reload
INFO: Will watch for changes in these directories: ['C:\\Users\\tully']
INFO: Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO: Started reloader process [336] using WatchFiles
INFO: Started server process [23472]
INFO: Waiting for application startup.
INFO: Application startup complete.

```

Ambiente Api (que foi gerado):

default

GET /clientes/ Ler Clientes

POST /clientes/ Cnlar Cliente

GET /clientes/{cliente_id} Ler Cliente

PUT /clientes/{cliente_id} Atualizar Cliente

DELETE /clientes/{cliente_id} Deletar Cliente

1. CRUD Completo:

Create (POST) - Validação de dados

Read (GET) - Com paginação

Update (PUT) - Atualização parcial

Delete (DELETE) - Confirmação de exclusão

EX:

POST /clientes/ Cnlar Cliente

Parameters

No parameters

Request body ^{required}

application/json

Edit Value | Schema

```
{
  "name": "Joana",
  "email": "joana.fi.iguaredo@gmail.com",
  "telefone": "(34)97777-8999",
  "empresa": "Boticario",
  "segmento": "perfume"
}
```

Execute Clear

Responses

Curl

```
curl -X 'POST' \
  http://localhost:8080/clientes/ \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "name": "Joana",
    "email": "joana.fi.iguaredo@gmail.com",
    "telefone": "(34)97777-8999",
    "empresa": "Boticario",
    "segmento": "perfume"
  }'
```

Request URL

http://localhost:8080/clientes/

Server response

Code Details

200

Response body

```
{
  "name": "Joana",
  "email": "joana.fi.iguaredo@gmail.com",
  "telefone": "(34)97777-8999",
  "empresa": "Boticario",
  "segmento": "perfume",
  "id": "6284ec3b7fac178b310fc31"
}
```

Response headers

```
content-length: 112
content-type: application/json
date: Thu, 07 Aug 2025 18:12:27 GMT
server: uvicorn
```

Responses

Code Description Links