

Documentação - Most Wanted

O projeto Most Wanted é um aplicativo Python que permite coletar informações sobre procurados internacionais da Interpol e do FBI e armazená-las em um banco de dados Oracle. O aplicativo interage com as APIs da Interpol e do FBI para obter informações atualizadas sobre os procurados e as armazena em uma tabela de banco de dados.

Requisitos

- Python 3.11
- Bibliotecas Python: [oracledb](#), [requests](#), [json](#)

```
pip install oracledb requests
```

Executando o Código

1. Execute o arquivo [most_wanted.py](#) para iniciar a coleta de dados:

```
python most_wanted.py
```

O aplicativo começará “dropando” a tabela T_PROCURADOS caso ela exista, criando novamente e em seguida coletará informações sobre procurados da Interpol e do FBI e as armazenará no banco de dados.

O aplicativo criará uma tabela chamada T_PROCURADOS no banco de dados Oracle. A tabela é formada pelas seguintes colunas:

- ID: (chave primária) gerado automaticamente.
- NOME: Nome do procurado.
- SOBRENOME: Sobrenome do procurado.
- IDADE: Idade do procurado (formato de texto, alguns registros armazenam data de nascimento, idade máxima, idade aproximada, etc. Grande parte dos registros do FBI não tem o campo idade preenchido, armazenando null na coluna).
- CRIME: Descrição do crime associado ao procurado.
- FONTE: Origem da informação (se é da API da Interpol ou do FBI).

Para a atividade limitamos em 100 registros no banco, sendo 50 da API do FBI e 50 da API da Interpol, o tempo de resposta é longo e pode gerar time-outs com requisições grandes. O número de registros pode ser alterado no código nos campos abaixo, e se removido, fará a busca pelos dados até o último registro armazenado pela Interpol e FBI.

```
max_registros_interpol = 50

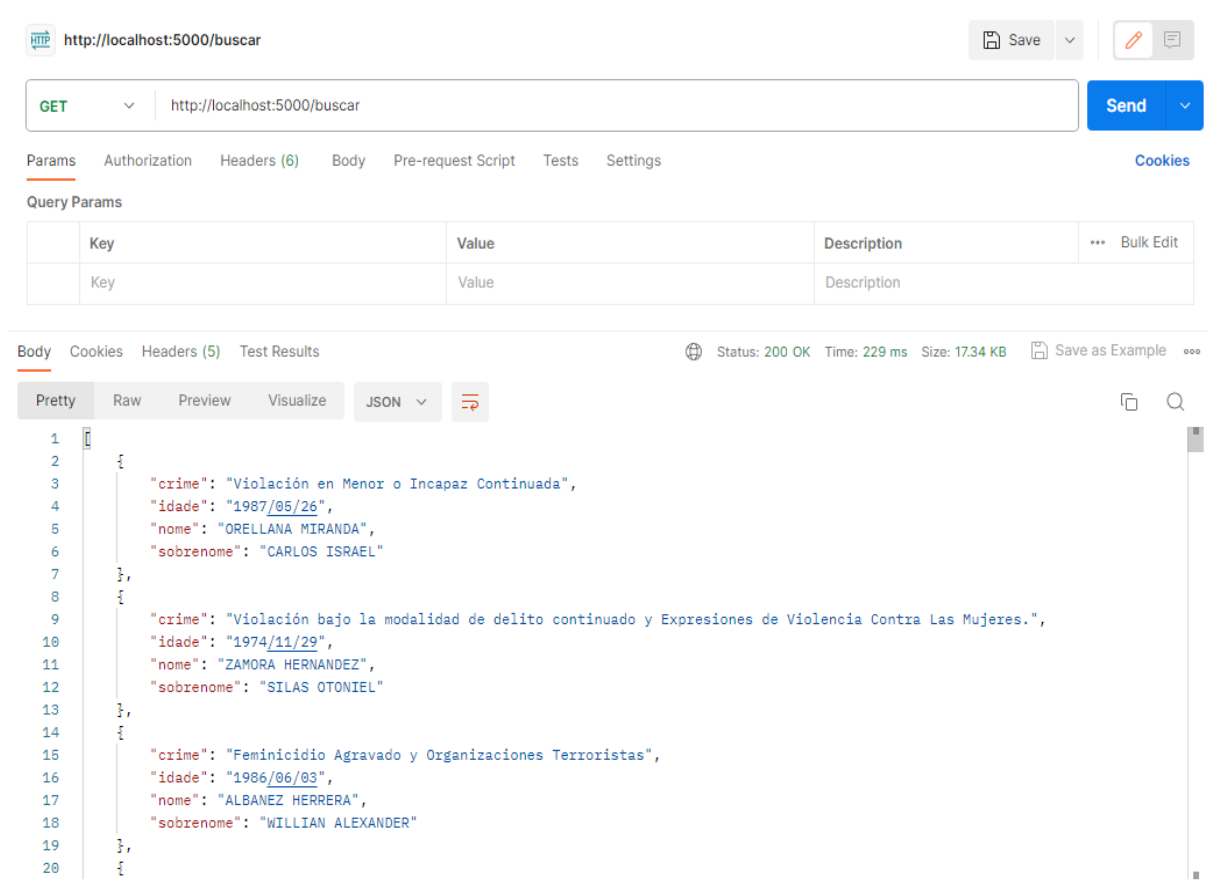
max_registros_fbi = 50
```

Endpoints da API Most Wanted

Além da coleta de dados, o projeto Most Wanted inclui uma API web para consultar os dados coletados. A API possui os seguintes endpoints:

Endpoint 1: /buscar

- Método: GET
- URL: <http://localhost:5000/buscar>
- Descrição: Retorna uma lista de todos os procurados armazenados no banco de dados.



HTTP <http://localhost:5000/buscar> Save

GET <http://localhost:5000/buscar> Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

Key	Value	Description	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (5) Test Results Status: 200 OK Time: 229 ms Size: 17.34 KB Save as Example

Pretty Raw Preview Visualize JSON

```
1 {
2   {
3     "crime": "Violación en Menor o Incapaz Continuada",
4     "idade": "1987/05/26",
5     "nome": "ORELLANA MIRANDA",
6     "sobrenome": "CARLOS ISRAEL"
7   },
8   {
9     "crime": "Violación bajo la modalidad de delito continuado y Expresiones de Violencia Contra Las Mujeres.",
10    "idade": "1974/11/29",
11    "nome": "ZAMORA HERNANDEZ",
12    "sobrenome": "SILAS OTONIEL"
13  },
14  {
15    "crime": "Feminicidio Agravado y Organizaciones Terroristas",
16    "idade": "1986/06/03",
17    "nome": "ALBANEZ HERRERA",
18    "sobrenome": "WILLIAN ALEXANDER"
19  },
20 }
```

Endpoint 2: /buscar_nome

- Método: GET
- URL: `http://localhost:5000/buscar_nome?nome=Jose`
- Parâmetros: `nome` (primeiro nome do procurado)
- Descrição: Retorna os detalhes de todos os procurados que tenham o mesmo nome, ignorando o sobrenome. No exemplo abaixo API retorna o registro de dois procurados com o nome “Jose”

The screenshot shows a REST client interface with the following components:

- URL Bar:** `http://localhost:5000/buscar_nome?nome=Jose`
- Method:** `GET`
- Send Button:** A blue button labeled "Send".
- Params Tab:** A table showing query parameters.
- Body Tab:** A section showing the response body in JSON format.

Query Params Table:

Key	Value	Description
<input checked="" type="checkbox"/> nome	Jose	

Response Body (JSON):

```
{
  "crime": "Murder in Aid of Racketeering (2 Counts)",
  "idade": null,
  "nome": "JOSE",
  "sobrenome": "SANTOS"
},
{
  "crime": "Murder in Aid of Racketeering (2 Counts)",
  "idade": null,
  "nome": "JOSE",
  "sobrenome": "ARMANDO"
}
```

Metadata: Status: 200 OK, Time: 222 ms, Size: 432 B

Endpoint 3: /buscar_nomecompleto

- Método: GET
- URL: `http://localhost:5000//buscar_nomecompleto?nome=jose&sobrenome=santos`
- Parâmetros: `nome`, `sobrenome`
- Descrição: Retorna o registro de um procurado a partir dos parâmetros nome e sobrenome. Seguindo o exemplo do endpoint 2, agora informando o sobrenome “santos”, apenas o registro do procurado Jose Santos é encontrado.

The screenshot shows an HTTP client interface with the following details:

- URL:** `http://localhost:5000//buscar_nomecompleto?nome=jose&sobrenome=santos`
- Method:** GET
- Query Params:**

Key	Value	Description
nome	jose	
sobrenome	santos	
- Response:**

```
1 {
2   "crime": "Murder in Aid of Racketeering (2 Counts)",
3   "idade": null,
4   "nome": "JOSE",
5   "sobrenome": "SANTOS"
6 }
```
- Status:** 200 OK, Time: 220 ms, Size: 300 B