API – EDP

**Resumo do Código**

O código tem a seguinte funcionalidade:

1. **Objetivo:**
   * Coleta de dados mensais ajustados da EDP Portugal usando a API Alpha Vantage.
   * Os dados incluem informações como: open, high, low, close, adjusted\_close, volume e dividend\_amount.
2. **Funcionamento:**
   * Envia uma solicitação para o endpoint da Alpha Vantage usando os parâmetros necessários (função, símbolo, chave da API).
   * Recebe os dados no formato JSON, extrai a série temporal (Monthly Adjusted Time Series) e converte para um **DataFrame do pandas**.
   * Salva os dados em um arquivo CSV no caminho especificado (C:\Users\isabe\Desktop\EDIT\EDIT\Data Ops\Company\_Project\data).
3. **Destaques Técnicos:**
   * Usa a biblioteca requests para acessar a API.
   * Usa o pandas para transformar os dados em um formato tabular e salvá-los como CSV.
   * Organiza as colunas de forma estruturada e ordena os dados pela data.

**Como Esconder a API Key**

Expor a API key diretamente no código não é seguro, especialmente se o código for compartilhado ou versionado (e.g., em um repositório GitHub). Aqui estão algumas maneiras de protegê-la:

**1. Usar Variáveis de Ambiente**

* Configure sua chave como uma variável de ambiente no sistema operacional.
* No Windows:

cmd

CopyEdit

set ALPHA\_VANTAGE\_API\_KEY=IITM7YMI9CMXL71J

* No código Python, acesse a chave usando os.environ:

python

CopyEdit

import os

API\_KEY = os.getenv("ALPHA\_VANTAGE\_API\_KEY")

**2. Salvar em um Arquivo .env**

* Instale a biblioteca python-dotenv:

bash

CopyEdit

pip install python-dotenv

* Crie um arquivo .env no diretório do projeto:

makefile

CopyEdit

ALPHA\_VANTAGE\_API\_KEY=IITM7YMI9CMXL71J

* No código Python, carregue a chave com dotenv:

python

CopyEdit

from dotenv import load\_dotenv

import os

# Carregar variáveis do arquivo .env

load\_dotenv()

API\_KEY = os.getenv("ALPHA\_VANTAGE\_API\_KEY")

**3. Armazenar em um Arquivo Configurado (e Excluí-lo do Git)**

* Crie um arquivo de configuração, como config.json:

json

CopyEdit

{

"api\_key": "IITM7YMI9CMXL71J"

}

* No código Python, carregue o arquivo:

python

CopyEdit

import json

with open("config.json", "r") as f:

config = json.load(f)

API\_KEY = config["api\_key"]

* Adicione o arquivo config.json ao .gitignore para que ele não seja versionado:

arduino

CopyEdit

config.json

**Recomendação**

A abordagem **.env com dotenv** é geralmente a mais prática para projetos Python, pois é fácil de configurar e segura se o arquivo .env for excluído do Git.

Se precisar de ajuda para implementar, é só avisar! 🚀

**Definir Variáveis Temporárias no PowerShell**

1. **Definir a Variável no Ambiente Atual**
   * Use o prefixo $env: para definir uma variável temporária:

powershell

CopyEdit

$env:API\_KEY = "IITM7YMI9CMXL71J"

1. **Verificar a Variável**
   * Para confirmar que a variável foi definida:

powershell

CopyEdit

echo $env:API\_KEY

1. **Usar no Código Python**
   * A variável estará disponível no ambiente enquanto a sessão do PowerShell estiver aberta. No Python, acesse-a usando os.environ:

python

CopyEdit

import os

# Obter a API Key

API\_KEY = os.environ.get("API\_KEY")

if API\_KEY:

print(f"A API Key é: {API\_KEY}")

else:

print("A API Key não foi definida.")

**Remover a Variável**

Se quiser remover a variável do ambiente, você pode usar:

powershell

CopyEdit

Remove-Item Env:API\_KEY

ATUALIZAR O README  
  
  
Aqui está um modelo de **README.md** focado na primeira parte do projeto, que coleta os dados históricos da API Alpha Vantage:

markdown

CopyEdit

# EDP Project

Este projeto tem como objetivo coletar e processar dados históricos da ação \*\*EDP Portugal (EDP.LS)\*\*, utilizando a API Alpha Vantage. Nesta primeira fase, estamos focados em obter os dados ajustados mensais e salvá-los localmente em formato CSV.

---

## ⚙️ Funcionalidades

- Conexão com a API \*\*Alpha Vantage\*\*.

- Coleta de dados históricos ajustados mensais da ação \*\*EDP Portugal\*\*.

- Armazenamento dos dados em um arquivo CSV.

---

## 📦 Estrutura do Projeto

EDP\_Project/ │ ├── src/ # Scripts principais │ └── fetch\_data.py # Código para coletar dados da API │ ├── data/ # Dados coletados (salvos aqui como CSV) │ ├── venv/ # Ambiente virtual (ignorado pelo Git) │ ├── .gitignore # Arquivos/diretórios ignorados pelo Git ├── README.md # Documentação do projeto

yaml

CopyEdit

---

## 🚀 Como Usar

### Pré-requisitos

1. \*\*Python 3.11 ou superior\*\* instalado.

2. Uma chave de API da [Alpha Vantage](https://www.alphavantage.co/).

### Passo a Passo

1. Clone este repositório:

```bash

git clone https://github.com/roasfora/EDP\_Project.git

cd EDP\_Project

1. Crie e ative um ambiente virtual:

bash

CopyEdit

python -m venv venv

# Ativar o ambiente virtual:

# No Windows:

.\venv\Scripts\activate

1. Instale as dependências:

bash

CopyEdit

pip install -r requirements.txt

1. Configure sua **API Key** da Alpha Vantage como uma variável de ambiente:

powershell

CopyEdit

$env:API\_KEY = "SUA\_API\_KEY"

1. Execute o script para coletar os dados:

bash

CopyEdit

python src/fetch\_data.py

1. Os dados serão salvos como um arquivo CSV em:

bash

CopyEdit

data/monthly\_adjusted\_data.csv

**📝 Exemplo de Saída**

Exemplo de um arquivo CSV gerado:

csv

CopyEdit

date,open,high,low,close,adjusted\_close,volume,dividend\_amount

2023-01-31,4.50,4.60,4.45,4.55,4.55,10000,0.02

2023-02-28,4.55,4.65,4.50,4.60,4.60,15000,0.01

...

**📚 Recursos Utilizados**

* [Alpha Vantage API](https://www.alphavantage.co/)
* Requests
* Pandas

**🌟 Próximos Passos**

* Implementar transformação dos dados coletados com DBT.
* Configurar a orquestração com Apache Airflow.
* Adicionar monitoramento com Grafana e Morpheu.
* Criar um dashboard interativo com Streamlit.

**🧑‍💻 Contribuição**

Contribuições são bem-vindas! Siga os passos abaixo:

1. Faça um fork do projeto.
2. Crie uma branch para sua funcionalidade:

bash

CopyEdit

git checkout -b minha-feature

1. Faça um commit das suas alterações:

bash

CopyEdit

git commit -m "Adiciona minha funcionalidade"

1. Envie para o repositório remoto:

bash

CopyEdit

git push origin minha-feature

1. Abra um Pull Request.

**📜 Licença**

Este projeto está licenciado sob a licença MIT.

yaml

CopyEdit

---

### \*\*Como Atualizar o README\*\*

- À medida que o projeto avança, você pode incluir as novas partes (transformação com DBT, monitoramento com Grafana, etc.).

- Caso precise de ajuda para editar futuramente, é só pedir! 🚀