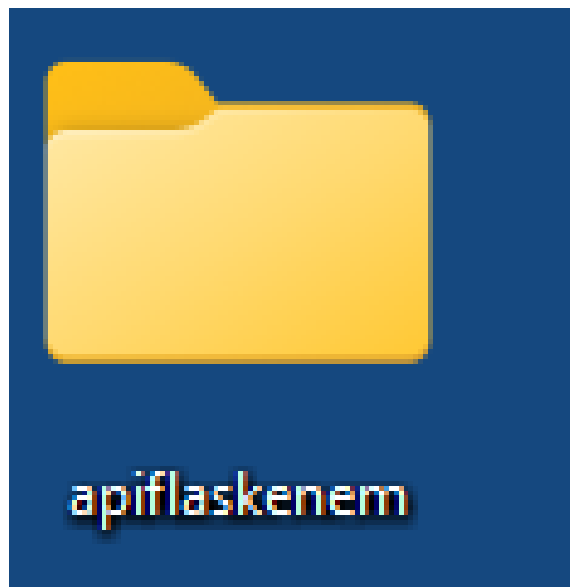
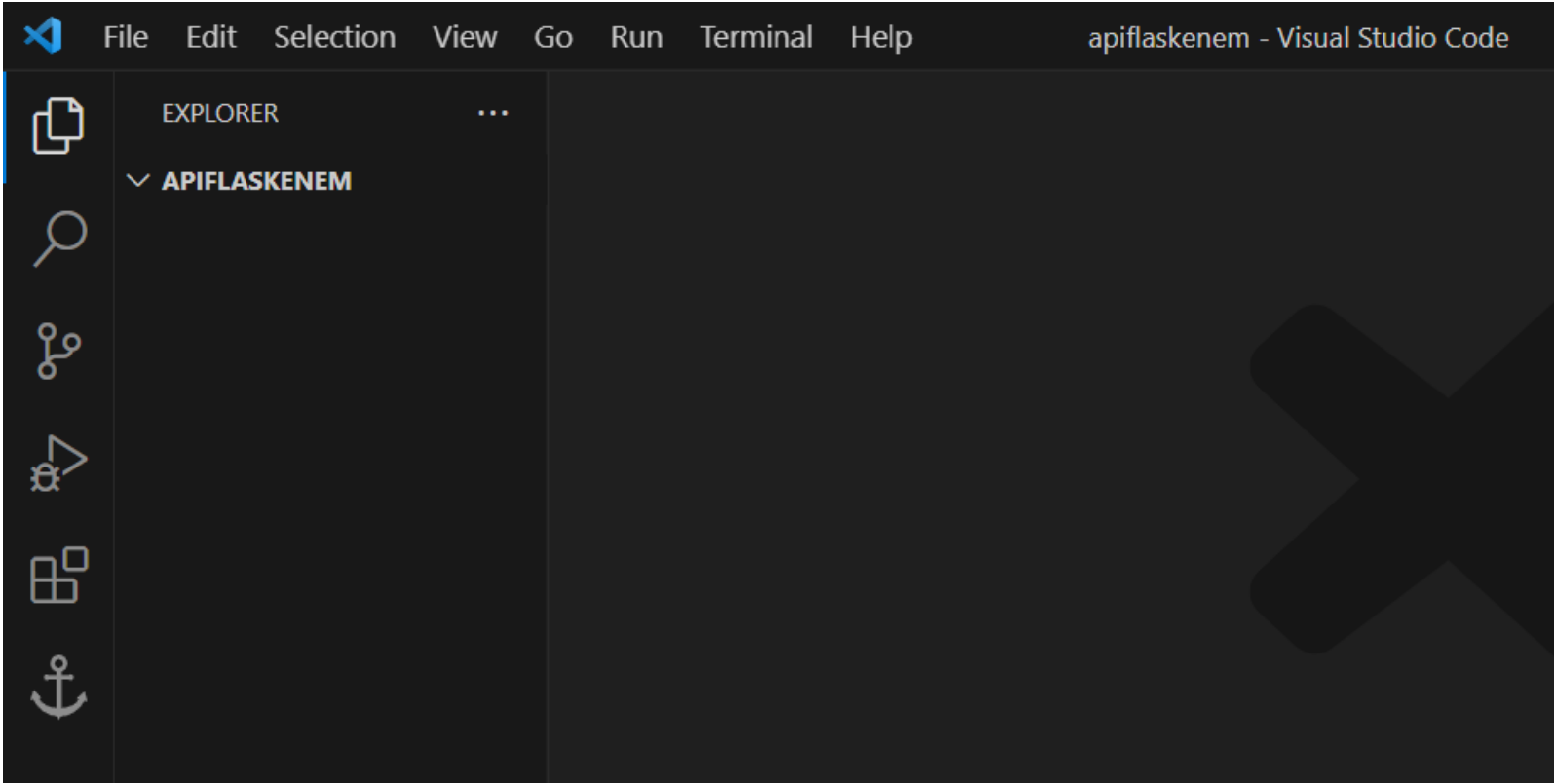


**Avançar : SIM ou NÃO**



Avançar : SIM ou NÃO



Avançar : SIM ou NÃO

PROBLEMS

OUTPUT

DEBUG CONSOLE


TERMINAL

COPILLOT VOICE

```
PS C:\Users\romulo.leite\Desktop\apiflaskenem> python -m venv venv
```

```
PS C:\Users\romulo.leite\Desktop\apiflaskenem> 
```

✓ APIFLASKENEM

>  venv


PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

COPILOT VOICE

 (venv) C:\Users\romulo.leite\Desktop\apiflaskenem\venv\Scripts>

```
(venv) C:\Users\romulo.leite\Desktop\apiflaskenem\venv\Scripts>cd ..
```

```
(venv) C:\Users\romulo.leite\Desktop\apiflaskenem\venv>cd ..
```

```
(venv) C:\Users\romulo.leite\Desktop\apiflaskenem>
```

```
(venv) C:\Users\romulo.leite\Desktop\apiflaskenem>pip install flask
```

```
Collecting flask
```

```
  Using cached flask-3.0.3-py3-none-any.whl.metadata (3.2 kB)
```

```
Collecting Werkzeug>=3.0.0 (from flask)
```

```
  Using cached werkzeug-3.0.2-py3-none-any.whl.metadata (4.1 kB)
```

```
Collecting Jinja2>=3.1.2 (from flask)
```

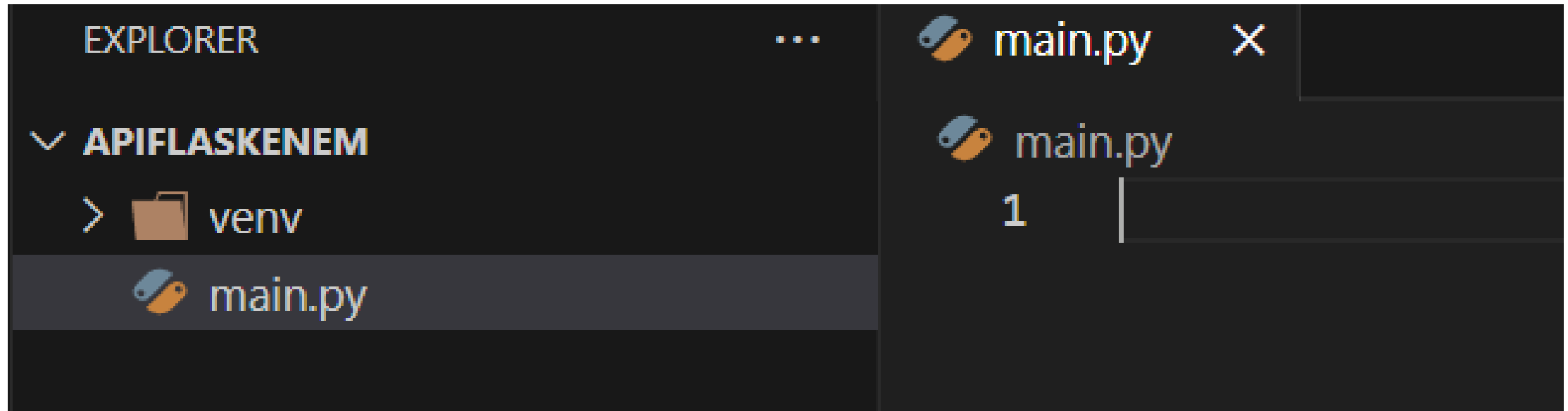
```
  Using cached Jinja2-3.1.3-py3-none-any.whl.metadata (3.3 kB)
```

```
Collecting itsdangerous>=2.1.2 (from flask)
```

# Framework

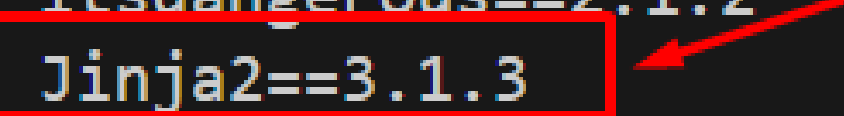
Conjunto de recursos (classes, métodos, recursos prontos)

# main.py



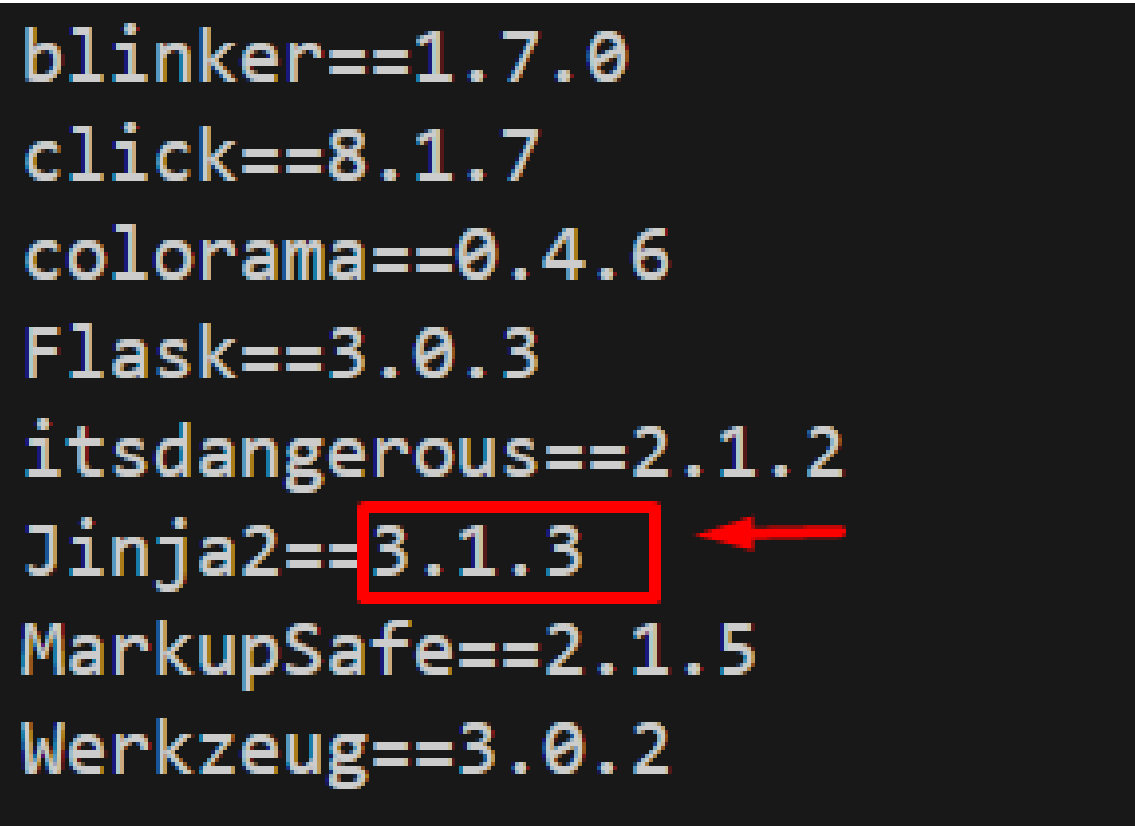


```
(venv) C:\Users\romulo.leite\Desktop\apiflaskenem>pip freeze  
blinker==1.7.0  
click==8.1.7  
colorama==0.4.6  
Flask==3.0.3  
itsdangerous==2.1.2  
Jinja2==3.1.3  
MarkupSafe==2.1.5  
Werkzeug==3.0.2
```





```
>pip freeze
```



```
blinker==1.7.0  
click==8.1.7  
colorama==0.4.6  
Flask==3.0.3  
itsdangerous==2.1.2  
Jinja2==3.1.3  
MarkupSafe==2.1.5  
Werkzeug==3.0.2
```

```
(venv) C:\Users\romulo.leite\Desktop\apiflaskenem>pip list
```

Package	Version
-----	-----
blinker	1.7.0
click	8.1.7
colorama	0.4.6
Flask	3.0.3
itsdangerous	2.1.2
Jinja2	3.1.3
MarkupSafe	2.1.5
pip	24.0
Werkzeug	3.0.2

EXPLORER

...

✓ APIFLASKENEM

> venv

main.py

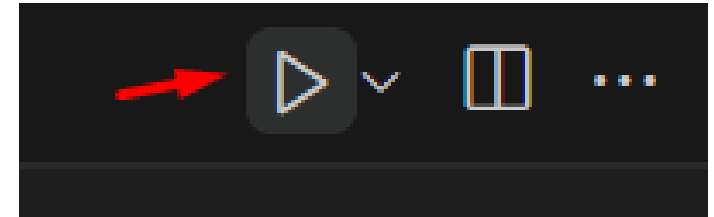
main.py ×

main.py > ...

```
1 #vou importar a classe Flask do pacote flask
2 from flask import Flask
3
4 #FIXME: Flask é um classe
5 #instanciando o objeto app
6 #Assume o nome da aplicação como nome padrão
7 app = Flask("__name__")
```

#executar a API

app.run()



```
8
9  #executar a API
10 app.run()
11
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COPILOT VOICE

erzeug 3.0.2

venv) C:\Users\romulo.leite\Desktop\apiflaskenem>

Ln 10, Col 10 Spaces: 4 UTF-8 CRLF { Python 3.12.3 ('venv': venv)

```
(venv) C:\Users\romulo.leite\Desktop\apiflaskenem>
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

COPilot VOICE

\* Running on `http://127.0.0.1:5000`

**Press CTRL+C to quit**



PROBLEMS

OUTPUT

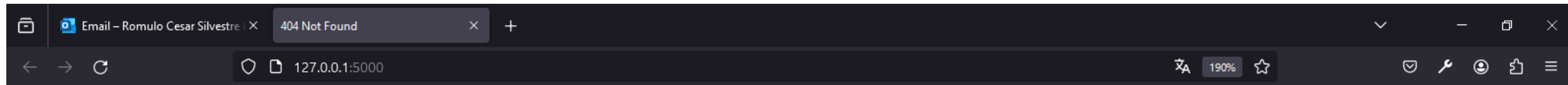
Follow link (ctrl + click)

TERMINAL

COPILOT VO

\* Running on <http://127.0.0.1:5000>

Press CTRL+C to quit



# Not Found


The requested URL was not found on the server. If you entered the URL manually please check your spelling and try again.

\* Running on http://127.0.0.1:5000

Press CTRL+C to quit

127.0.0.1 - - [15/Apr/2024 20:20:48] "GET / HTTP/1.1" 404 -

PS C:\Users\romulo.leite\Desktop\apiflaskenem> █

 dados.py > ...

```
3  #Criar uma lista
4  #lista e tupla - dicionário ???
5
6  materias_enem = [
7      {"id":1,"nome":"Matemática"},
8      {"id":2,"nome": "Linguagens,códigos e suas tecnologias"},
9      {"id":3,"nome": "Ciências Humanas"},
10     {"id":4,"nome": "Ciências da Natureza"}
11 ]
```

main.py > ...

```
1  #vou importar a classe Flask do pacote flask
2  from flask import Flask
3  from dados import Materias
4
5  #FIXME: Flask é um classe
6  #instanciando o objeto app
7  #Assume o nome da aplicação como nome padrão
8  app = Flask("__name__")
9  #NOTE: vamos decorar
10 #exemplo sem decorar
11 #será que o flask consegue entender essa função solta ai?
12
13 def get_materias():
14     return Materias
15
16
17
18 #executar a API
19 app.run()
20
```



**@admantium**

```
def usar_maos(self):  
    pass
```



127.0.0.1:5000

```
@app.route('/materias', methods=['GET'])  
def get_materias():
```

← → ↻ 127.0.0.1:5000/materias

JSON Dados brutos Cabeçalhos

Salvar Copiar Recolher tudo Expandir tudo Filtrar JSON

▼ 0:

- id: 1
- nome: "Matemática"

▼ 1:

- id: 2
- nome: "Linguagens, códigos e suas tecnologias"

▼ 2:

- id: 3
- nome: "Ciências Humanas"

▼ 3:

- id: 4
- nome: "Ciências da Natureza"





# Como testar uma API usando uma ferramenta profissional.

- Utilizaremos o POSTMAN que é uma ferramenta utilizada para testar a API.



Product ^

Pricing

Enterprise

Resources and Support v

Public API Network

### Getting Started

What is Postman?

Customer Stories

[Download Postman→](#)

### API Platform

Collaborate in Workspaces

Organize with Collections

Explore the API Client

Build Postman Flows

Work smarter with Postbot

Browse API Tools

### Enterprise Solution

Enterprise Essentials

API Test Automation

Internal API Management

Home

Notion

Collection

Environment

APIs

Mock Servers

Monitors

Alpha

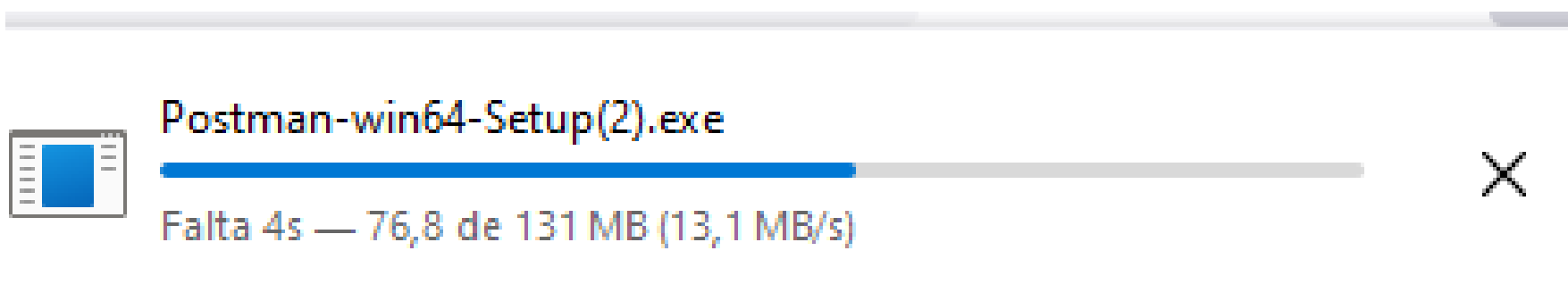
APIs together

## The Postman app


Download the app to get started with the Postman API Platform.

 Windows 64-bit

By downloading and using Postman, I agree to the [Privacy Policy](#) and [Terms](#).



[Mostrar todos os downloads](#)

GET  https://c

GET

POST

PUT

PATCH

DELETE

HEAD

OPTIONS

---

Type a new method

pegar

GET

http://127.0.0.1:5000/materias

Send

Params Authorization Headers (7) Body Pre-request Script Tests ● Settings

Cookies

## Query Params

	Key	Value	Description	...	Bulk Edit
	Key	Value	Description		

Body Cookies Headers (5) Test Results (0/1)

🌐 Status: 200 OK Time: 5 ms Size: 347 B 📄 Save as example ⋮

Pretty

Raw

Preview

Visualize

JSON



```
1  [
2    {
3      "id": 1,
4      "nome": "Matemática"
5    },
6    {
7      "id": 2,
8      "nome": "Linguagens, códigos e suas tecnologias"
9    },
10   {
11     "id": 3,
12     "nome": "Ciências Humanas"
13   },
14   {
15     "id": 4,
16     "nome": "Ciências da Natureza"
17   }
18 ]
```