

11 de Abril

26 de Maio

15 de Maio

26 de Maio

15 de Junho

26 de Junho

Apresentação da Documentação

Flet Android

Api Flask

Sistema Django

Power Point

templatepi.docx

pimodelo.docx

ABNT

Ortografia

Gramática

Logica de Programação com Python

Design System

Versionamento de Código da Equipe

Storytelling

Desenvolvimento Ágil e Entrega Contínua (SCRUM)

Diagramas: Objetos, Sequência e Atividades

Diagrama de Classes

Dicionário de Dados

Fluxogramas (Dia)

ER    DER    MER

Documento Textual de Caso de Uso

Diagrama de Caso de Uso

Cronograma I

Especificação de Requisitos    User History

Levantamento de Requisitos

Deploy (Desktop, Android, Web, API)

Microsoft Azure (Nuvem)

Sistema Web (Django)

Web API (Flask)

Flet (Desktop e Android)

SQLAlchemy

Orientação a Objetos com Python

Lógica de Programação com Python

Design System

Versionamento de Código da Equipe

Storytelling

Desenvolvimento Ágil e Entrega Contínua (SCRUM)

Diagramas: Objetos, Sequência e Atividades

Diagrama de Classes

Dicionário de Dados

Deploy (Desktop, Android, Web, API)

Microsoft Azure (Nuvem)

Sistema Web (Django)

Web API (Flask)

Flet (Desktop e Android)

SQLAlchemy

Orientação a Objetos com Python

Lógica de Programação com Python

Design System

Versionamento de Código da Equipe

Storytelling

Desenvolvimento Ágil e Entrega Contínua (SCRUM)

Diagramas: Objetos, Sequência e Atividades

Diagrama de Classes

Dicionário de Dados

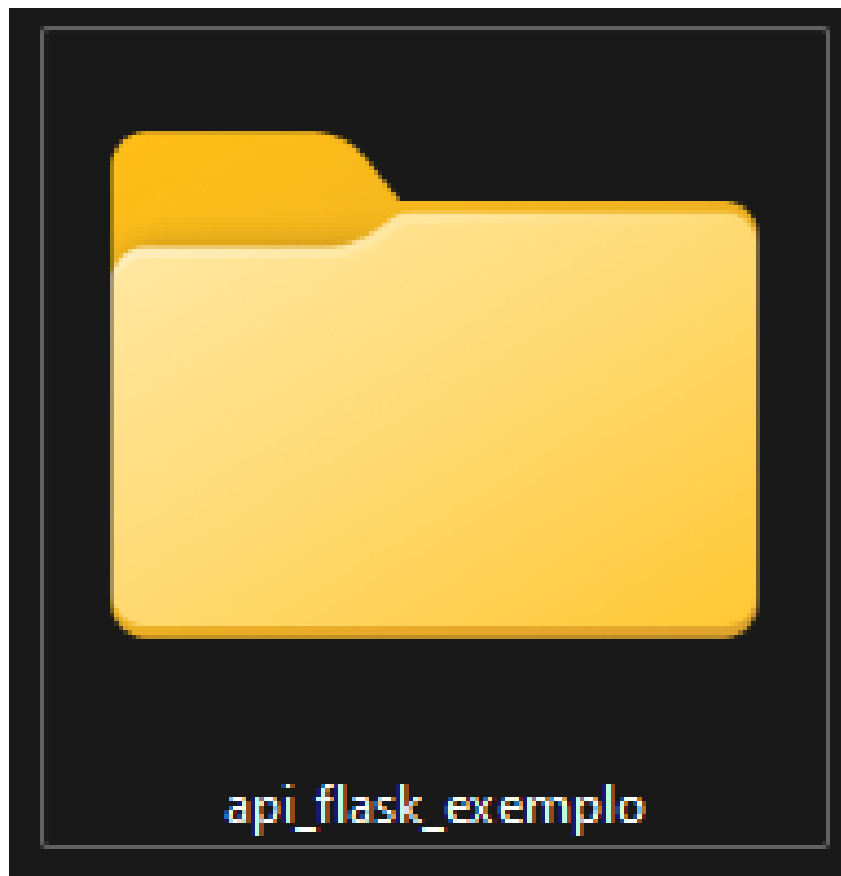


# Flask

Professor Rômulo Cesar Silvestre

# Ambientação

- Crie uma pasta: apiflaskexemplo



Ambiente Virtual

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

```
PS D:\projetointegrador\api_flask_exemplo> python -m venv venv
```



EXPLORER



✓ API\_FLASK\_EXEMPLO

>  venv

```
PS D:\projetointegrador\api_flask_exemplo> python -m venv venv
PS D:\projetointegrador\api_flask_exemplo> pip install flask
```

```
PS D:\projetointegrador\api_flask_exemplo> python -m venv venv
PS D:\projetointegrador\api_flask_exemplo> pip install flask
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: flask in c:\users\user\appdata\local\packages\pythonsoftwarefoundation.python.3.12_qbz5n2kfra8p0\localcache\local-packages\python312\site-packages (3.0.3)
Requirement already satisfied: Werkzeug>=3.0.0 in c:\users\user\appdata\local\packages\pythonsoftwarefoundation.python.3.12_qbz5n2kfra8p0\localcache\local-packages\python312\site-packages (from flask) (3.0.2)
Requirement already satisfied: Jinja2>=3.1.2 in c:\users\user\appdata\local\packages\pythonsoftwarefoundation.python.3.12_qbz5n2kfra8p0\localcache\local-packages\python312\site-packages (from flask) (3.1.3)
Requirement already satisfied: itsdangerous>=2.1.2 in c:\users\user\appdata\local\packages\pythonsoftwarefound
```

Ative sua máquina virtual

```
Microsoft Windows [versão 10.0.22631.3447]  
(c) Microsoft Corporation. Todos os direitos reservados.  
  
(venv) D:\projetointegrador\api_flask_exemplo>
```


EXPLORER




✓ API\_FLASK\_EXEMPLO

>  venv

 main.py

 main.py ×

 main.py

1



```
(venv) D:\projetointegrador\api_flask_exemplo>pip freeze
```

```
(venv) D:\projetointegrador\api_flask_exemplo>
```

```
(venv) D:\projetointegrador\api_flask_exemplo>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)
```

```
  Using cached itsdangerous-2.1.2-py3-none-any.whl.metadata (2.9 kB)
```


```
Collecting click>=8.1.3 (from flask)
```

```
  Using cached click-8.1.7-py3-none-any.whl.metadata (3.0 kB)
```

```
Collecting blinker>=1.6.2 (from flask)
```

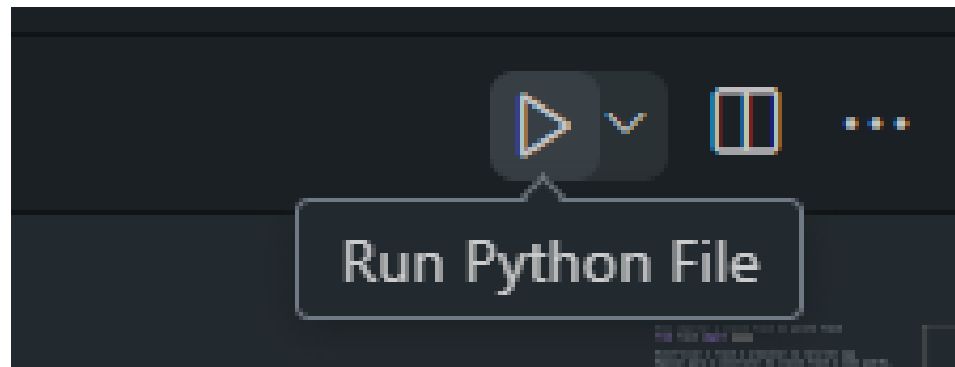
```
(venv) D:\projetointegrador\api_flask_exemplo>pip freeze  
blinker==1.7.0  
click==8.1.7  
colorama==0.4.6  
Flask==3.0.3  
itsdangerous==2.1.2  
Jinja2==3.1.3  
MarkupSafe==2.1.5  
Werkzeug==3.0.2
```



 main.py > ...

```
1  #vou importar a classe Flask do pacote flask
2  from flask import Flask
3
4  #instânciar o flask e armazenar na variável app
5  #passar para o construtor da classe flask o nome padrão.
6  #Ele assume o nome da aplicação como nome padrão.
7  app = Flask("__name__")
8
9  #rodar a api
10 app.run()
```

## Executando a Aplicação



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

**WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.**

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

Press CTRL+C to quit

pegou a porta 5000 - clique no link irá abrir no navegador

+ v ... ^ x

powersh... ⚠

cmd

Python

ambiente virtual



Ln 2, Col 24

Spaces: 4

UTF-8

CRLF



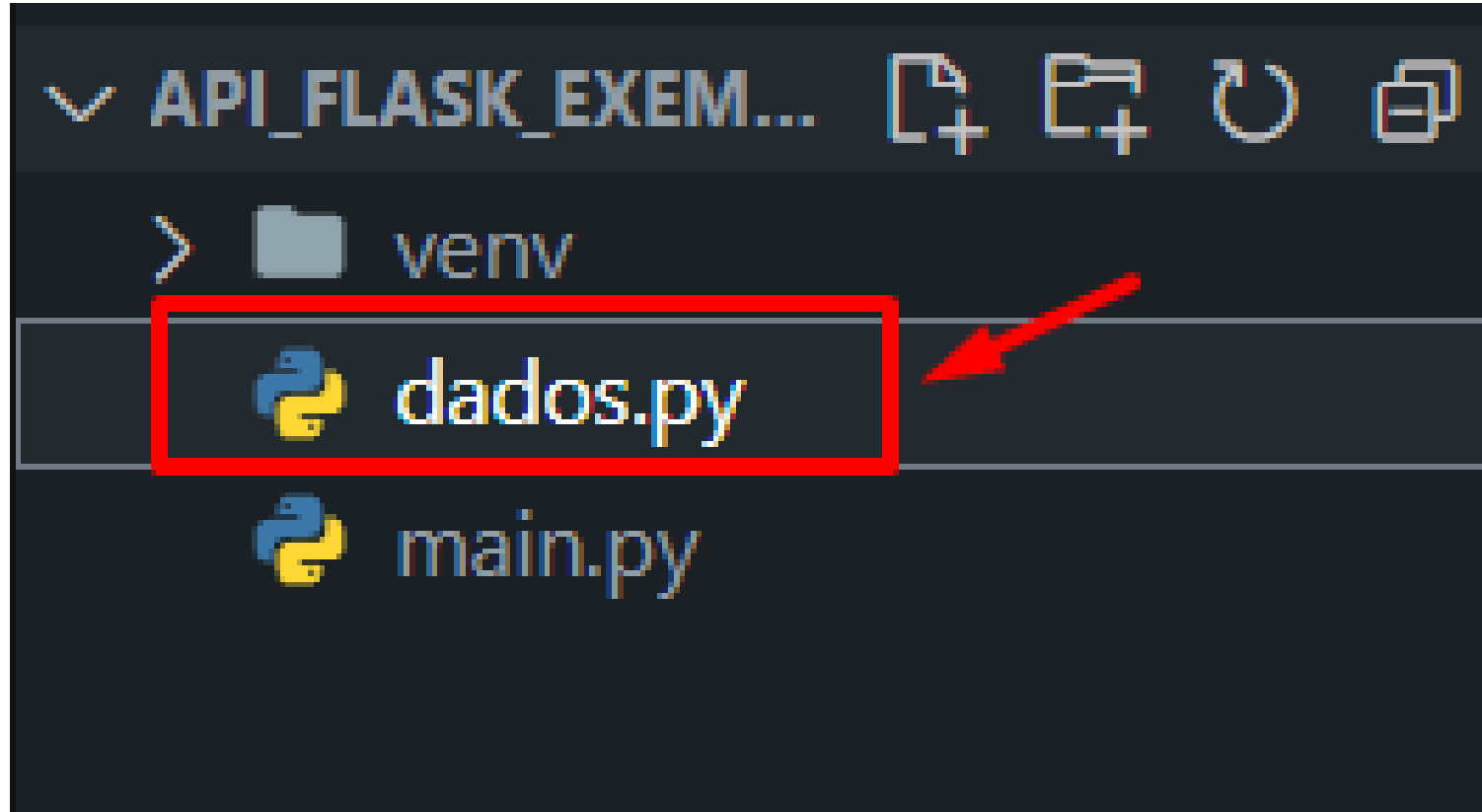
Python 3.12.2 ('venv': venv)



# Not Found

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

**CTRL+C (encerra o servidor)**



✓ API\_FLASK\_EXEMPLO

> venv

dados.py

main.py

dados.py > ...

```
1 # Criando a lista de matérias do ENEM
2 materias_enem = [
3     {"id": 1, "nome": "Matemática"},
4     {"id": 2, "nome": "Ciências da Natureza"},
5     {"id": 3, "nome": "Linguagens, Códigos e suas Tecnologias"},
6     {"id": 4, "nome": "Ciências Humanas e suas Tecnologias"}
7 ]
```

```
8  app = Flask("__name__")
9
10
11
12  #rodar a api
13  app.run()
```

Aqui podemos colocar nossas funções.  
Utilizar decorators

Decorators marca a função para alguma finalidade.



```
app = Flask("__name__")
```

```
def get_materias():  
    return Materias
```

O flask não entende  
essa função solta ai!

```
#rodar a api
```

"Decorators no Python são como as habilidades extras que o Wolverine ganha quando coloca suas garras de adamantium. Eles fortalecem suas funções ou classes, permitindo que realizem tarefas adicionais sem alterar sua essência."

```
@admantium  
def maos():  
    pass
```

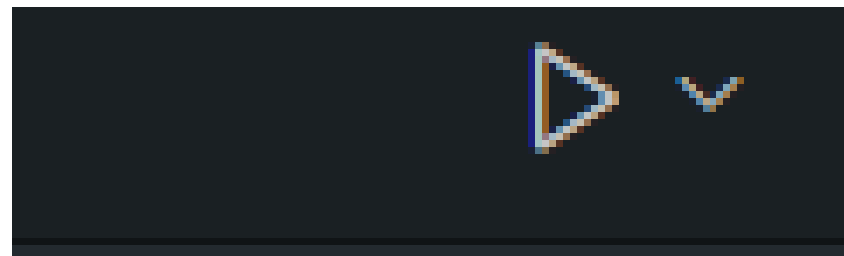


# @Decorators

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

```
@app.route('/materiais', ['GET'])  
def get_materias():  
    return Materias
```

8	app = Flask("__name__")
9	
10	@app.route('/materias', methods=['GET'])
11	def get_materias():
12	return Materias
13	#rodar a api
14	app.run()



\* Debug mode: off

WARNING: This is not a production server. Do not use  
I server instead

Follow link (ctrl + click)

\* 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.



Google

postman



Todas

Imagens

Videos

Shopping

Notícias

⋮ Mais

Aproximadamente 112.000.000 resultados (0,25 segundos)



Postman

<https://www.postman.com> · Traduzir esta página ⋮

Postman API Platform | Sign Up for Free



Product ^

Pricing

Enterprise

Resources and Support v

Public API Ne

## 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

APIs together

# The Postman app

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

 Windows 64-bit



Postman-win64-Setup.exe

Falta 1min 15s — 11,6 de 131 MB (2,6 MB/s)





# Join 25 million developers who use Postman

Simplify your API design, development and testing workflows with Postman.

- ⚡ Save and organize your work in collections and workspaces
- ⚡ Share your work with clients and collaborate with teammates
- ⚡ Integrate with tools like VS Code, GitHub and BitBucket

## Create a free Postman account

Work email

Create Free Account

Already have an account? [Log In](#)

Or continue with the [lightweight API client](#).

**Este site está tentando abrir o Postman.**

https://identity.getpostman.com deseja abrir este aplicativo.

☐ Sempre permitir identity.getpostman.com abrir links desse tipo no aplicativo associado

Abrir

Cancelar



**It's great to have you aboard,  
Rômulo Cesar Silvestre Leite**

Redirecting you to the Desktop App app

If you aren't redirected automatically, [use  
authorization token to sign in.](#)

Segurança do Windows

## Deseja permitir que redes públicas e privadas acessem este aplicativo?

O Firewall do Windows bloqueou alguns recursos de Postman em todas as redes públicas ou privadas.



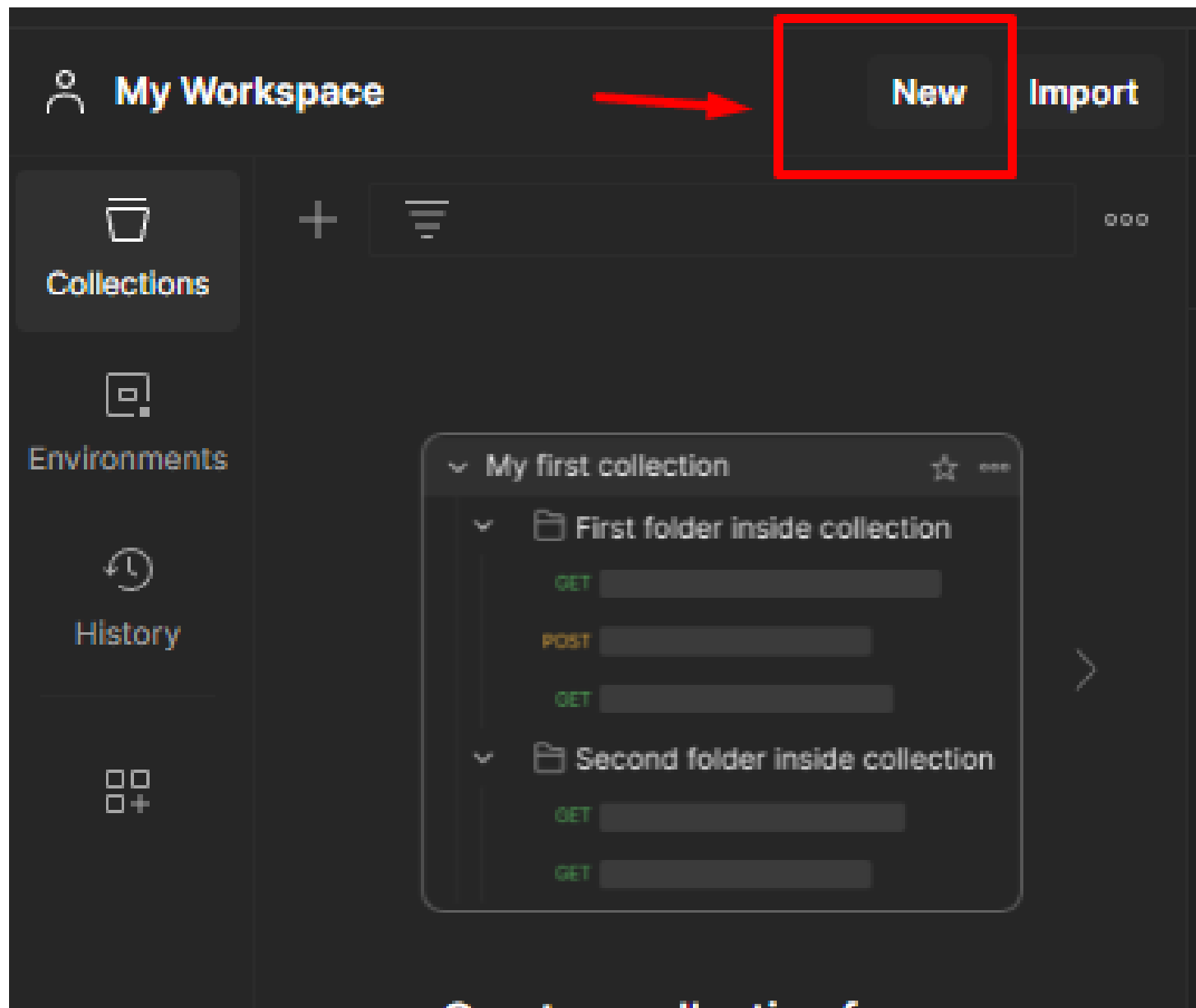
**Postman**

Fornecedor Postman

[Mostrar mais](#)

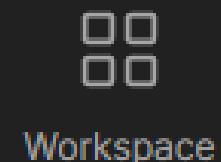
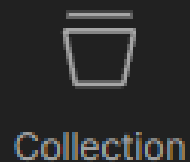
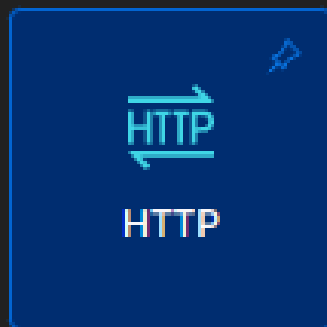
Permitir

Cancelar



Create a collection for your





Hypertext Transfer Protocol (HTTP) is an application-layer protocol often used to build REST APIs. Test your HTTP API with an HTTP request.



Untitled Request



Save



GET



Enter URL or paste text

Send



Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Cookies

### Query Params

	Key	Value	Description	...	Bulk Edit
	Key	Value	Description		

### Response



Enter the URL and click Send to get a response

WARNING: This is not a production server. Do not use it in production.  
I server instead

Follow link (ctrl + click)

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

Press CTRL+C to quit

127.0.0.1 - - [11/Apr/2024 15:41:39] "GET / HTTP/1.1"



127.0.0.1:5000

HTTP http://127.0.0.1:5000/

Save

selecione o método HTTP

GET

http://127.0.0.1:5000/

digite ou copie e cole a url

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: 404 NOT FOUND

Time: 17 ms

Size: 388 B



Save as example



Pretty

Raw

Preview

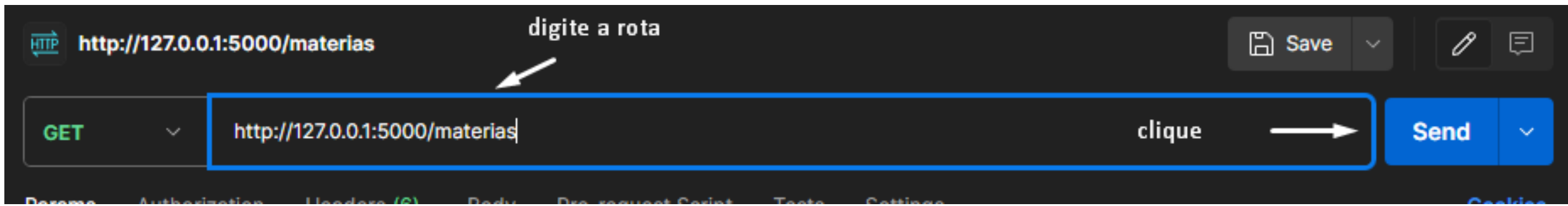
Visualize

HTML



```
1 <!doctype html>
2 <html lang=en>
3 <title>404 Not Found</title>
4 <h1>Not Found</h1>
5 <p>The requested URL was not found on the server. If you entered the URL manually please check your spelling and try
  again.</p>
6
```

Onde quero dar um **get**?





http://127.0.0.1:5000/materias



Save



GET



http://127.0.0.1:5000/materias

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: 12 ms

Size: 367 B



Save as example



Pretty

Raw

Preview

Visualize

JSON



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

```
from flask import Flask, make_response
```

O `make_response` geralmente gera em JSON



```
from flask import Flask, make_response, jsonify
```

```
10 @app.route('/materias', methods=['GET'])
11 def get_materias():
12     return make_response(
13         Materias
14     )
15 #rodar a api
16 app.run()
```

```
2  Materias = [  
3      {"id": 1, "nome": "Matemática"},  
4      {"id": 2, "nome": "Ciências da Natureza"},  
5      {"id": 3, "nome": "Linguagens, Códigos e suas Tecnologias"},  
6      {"id": 4, "nome": "Ciências Humanas e suas Tecnologias"}  
7  ]
```

Lista com Dicionários

```
11 def get_materias():
12     return make_response(
13         jsonify(
14             Materias
15         )
16     )
```



```
11 def get_materias():  
12     return make_response(  
13         jsonify(Materias)  
14     )
```



http://127.0.0.1:5000/materias



Save



GET



http://127.0.0.1:5000/materias

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: 13 ms

Size: 367 B



Save as example



Pretty

Raw

Preview

Visualize

JSON




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

Nosso *primeiro endpoint* a gente nunca esquece

**/materias - GET**

## Vamos avancar?

```
2 from flask import Flask, make_response, jsonify, request
3 #importando a lista de materias do enem
4 from materias import Materias
5 #instanciar o flask e armazenar na variável app
6 #passar para o construtor da classe flask o nome padrão.
7 #Ele assume o nome da aplicação como nome padrão.
8 app = Flask("__name__")
9
```





```
@app.route('/materiais', methods=['POST'])
def create_materia():
    materia = request.json
    return materia
```

GET



http://127.0.0.1:5000/materias

Send



GET

POST

PUT

PATCH

DELETE

HEAD

OPTIONS

Type a new method

Headers (6)

Body

Pre-request Script

Tests

Settings

Cookies

Value

Description

... Bulk Edit

Value

Description

5) Test Results



Status: 200 OK

Time: 13 ms

Size: 367 B



Save as example



View

Visualize

JSON



"Matemática"



http://127.0.0.1:5000/materias

Save



POST

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 (6)

Test Results



Status: 405 METHOD NOT ALLOWED

Time: 9 ms

Size: 370 B



Save as example



Pretty

Raw

Preview

Visualize

HTML



```
1 <!doctype html>
2 <html lang=en>
3 <title>405 Method Not Allowed</title>
4 <h1>Method Not Allowed</h1>
5 <p>The method is not allowed for the requested URL.</p>
```

vai dar um erro!!!!

# 405



O erro HTTP 405, também conhecido como "Method Not Allowed" (Método Não Permitido), ocorre quando um cliente tenta acessar um recurso em um servidor usando um método HTTP que não é permitido para esse recurso. Em outras palavras, o servidor recebeu uma solicitação com um método (por exemplo, GET, POST, PUT, DELETE) que não é suportado para o recurso solicitado.

**POST**



http://127.0.0.1:5000/materias

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

☐ GraphQL

JSON



1

# BUG

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

- URL:** `http://127.0.0.1:5000/materias`
- Method:** `POST`
- Body (JSON):**

```
{  1  {  2    "id": 5,  3    "nome": "redação"  4  }
```
- Response Status:** `405 METHOD NOT ALLOWED`
- Response Time:** `8 ms`
- Response Size:** `370 B`
- Response Content:**

## Method Not Allowed

The method is not allowed for the requested URL.

WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.

\* Running on http://127.0.0.1:5000

Press CTRL+C to quit

127.0.0.1 - - [11/Apr/2024 16:42:22] "POST /materias HTTP/1.1" 405 -







http://127.0.0.1:5000/materias

POST

http://127.0.0.1:5000/materias

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

☐ GraphQL

JSON

```
1 {  
2   ... "id": 5,  
3   ... "nome": "redação"  
4 }
```

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

JSON

```
1 {  
2   "id": 5,  
3   "nome": "redação"  
4 }
```



Status: 200 OK

Time: 8 ms

S

Body Cookies Headers (5) Test Results

Pretty

Raw


Preview

Visualize

JSON



```
12     "nome": "Linguagens, Códigos e suas Tecnologias"
13   },
14   {
15     "id": 4,
16     "nome": "Ciências Humanas e suas Tecnologias"
17   },
18   {
19     "id": 5,
20     "nome": "redação"
21   }
22 ]
```



POST



http://127.0.0.1:5000/materias

Send



Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cookies



none



form-data



x-www-form-urlencoded



raw



binary



GraphQL

JSON



Beautify

1



2

```
... "id":6,
```

3

```
... "nome":"educação física"
```

4



```
#0 Flask automaticamente ordena de A - Z alfabeticamente  
#para desabilitar isso , por padrão é True  
app.config['JSON_SORT_KEYS'] = False
```

```
@app.route('/materias', methods=['POST'])
def create_materia():
    materia = request.json
    Materias.append(materia)
    return make_response(
        jsonify(
            materia
        )
    )
```

# Tudo pode melhorar!!!!

```
@app.route('/materias', methods=['GET'])
def get_materias():
    return make_response(
        jsonify(
            mensagem = "Lista de Materias",
            dados = Materias
        )
    )
```

# Vamos conectar com o banco de dados?

Modelo Conceitual

Modelo Lógico

Modelo Físico

Engenharia Reversa – Code/Model – Entidade do MySQL WorkBench

Conexão com Banco

Modelagem UML

Caso de Uso

Sequência

Atividade

Classes

Objetos





## materia

 id: int

nome: varchar(95)



 Editor



12



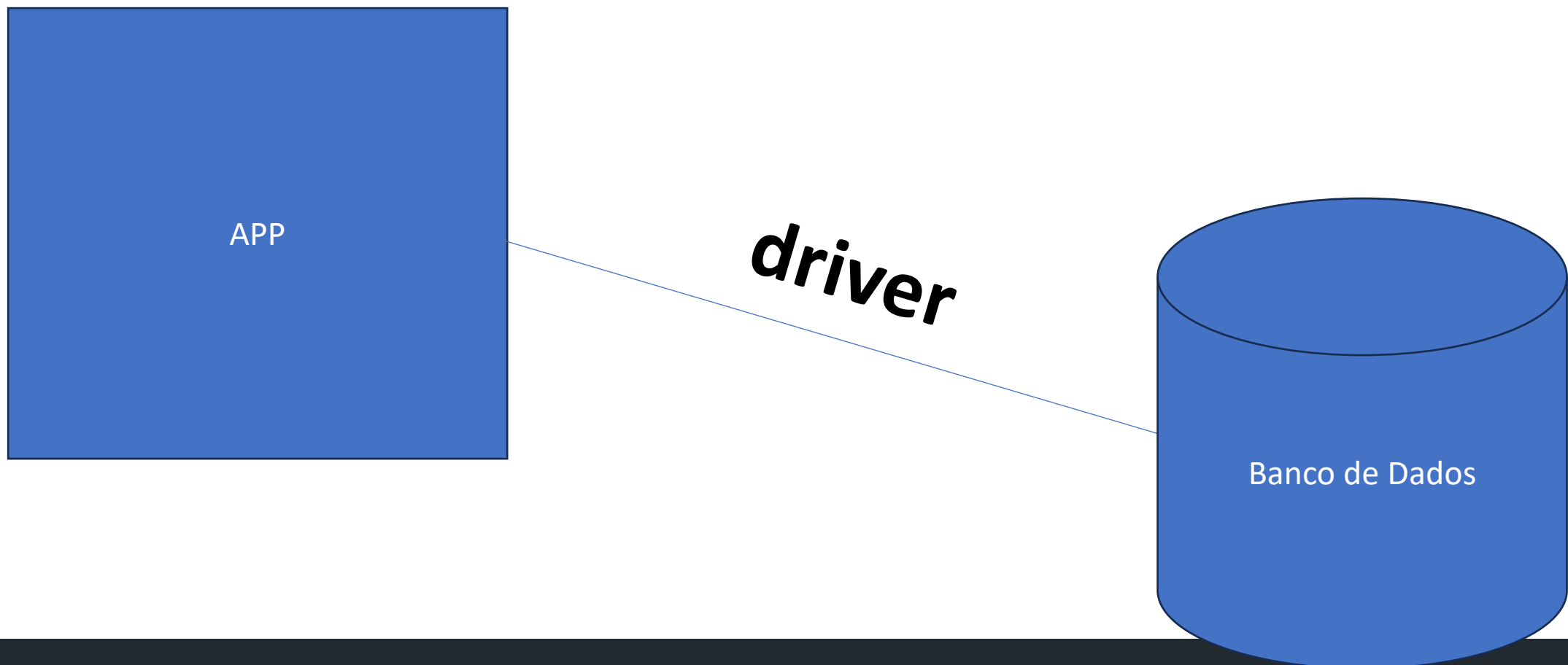
```
1  /* Lógico_1: */  
2  
3  CREATE TABLE materia (  
4      id int PRIMARY KEY auto_increment,  
5      nome varchar(95) NOT NULL  
6  );  
7  |
```

1 • **CREATE DATABASE** enem;

2 • **USE** enem;

3 • **CREATE TABLE** materia (  
4       id **int PRIMARY KEY auto\_increment**,  
5       nome **varchar(95)NOT NULL**  
6    );  
7

```
1 • CREATE DATABASE enem;
2 • USE enem;
3 • CREATE TABLE materia (
4     id int PRIMARY KEY auto_increment,
5     nome varchar(95)NOT NULL
6 );
7 • INSERT INTO materia(nome)values('português');
8 • INSERT INTO materia(nome)values('matematica');
9 • SELECT * FROM materia;
10
```



```
PS D:\projetointegrador\api_flask_exemplo> pip install mysql-connector-python
Collecting mysql-connector-python
  Downloading mysql_connector_python-8.3.0-cp312-cp312-win_amd64.whl.metadata (2.0 kB)
  Downloading mysql_connector_python-8.3.0-cp312-cp312-win_amd64.whl (15.4 MB)
  

8.3/15.4 MB 2.8 MB/s eta 0:00:03


```

```
main.py > ...  
1  #vou importar a classe Flask do pacote flask  
2  from flask import Flask,make_response,jsonify,request  
3  #importando a lista de materias do enem  
4  from materias import Materias  
5  #instanciar o flask e armazenar na variável app  
6  #passar para o construtor da classe flask o nome padrão  
7  #ele assume o nome da aplicação como nome padrão.  
8  import _mysql_connector  
9  app = Flask("__name__")  
10
```

```
12 mydb = mysql.connector.connect(  
13     host = 'localhost',  
14     user = 'root',  
15     password= 'root',  
16     database = 'enem',  
17 )
```

```
#vou importar a classe Flask do pacote flask
from flask import Flask,make_response,jsonify,request
#importando a lista de materias do enem
from materias import Materias
#instanciar o flask e armazenar na variável app
#passar para o construtor da classe flask o nome padrão.
#Ele assume o nome da aplicação como nome padrão.
import mysql.connector
#conexão com banco de dados
```



```
app = Flask("__name__")

#0 Flask automaticamente ordena de A - Z alfabeticamente
#para desabilitar isso , por padrão é True
app.config['JSON_SORT_KEYS'] = True

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

    #cursor é um agente para execução de SQL
    mycursor = mydb.cursor()
    mycursor.execute('SELECT * FROM materia')
    minhas_materias = mycursor.fetchall() #retorna a lista de todos os dados que ele conseguiu
    print(minhas_materias)
    return make_response(
        jsonify(
            mensagem = "lista de matérias cadastradas",
            dados = minhas_materias
        )
    )
```

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

    #cursor é um agente para execução de SQL
    mycursor = mydb.cursor()
    mycursor.execute('SELECT * FROM materia')
    minhas_materias = mycursor.fetchall() #retorna a lista de todos os dados que ele
    print(minhas_materias)
    return make_response(
        jsonify(
            mensagem = "lista de matérias cadastradas",
            dados = minhas_materias
        )
    )
```