Dockers e Containers

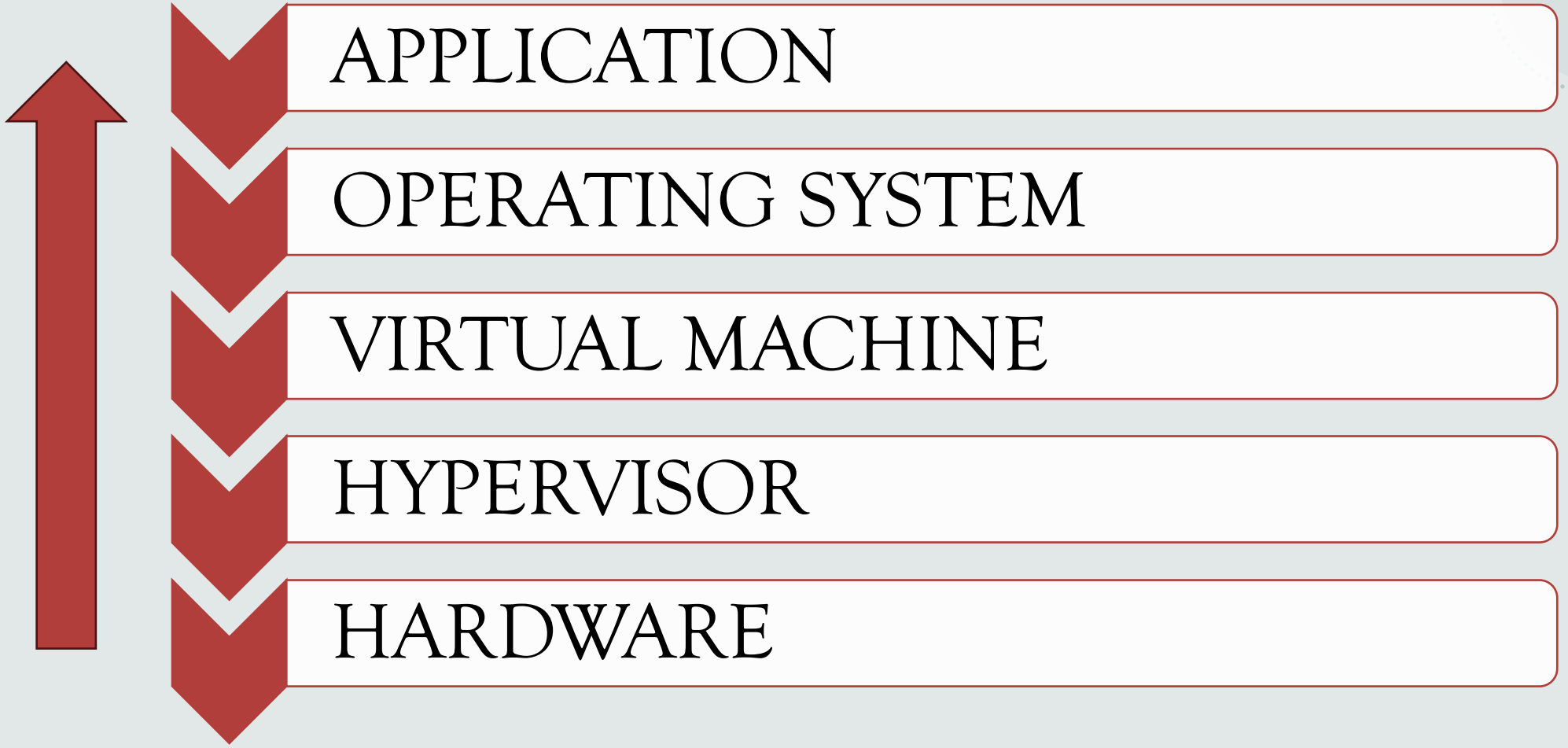
CNW-I

PROF. ANDERSON VANIN

Situação Problema

Rodar uma aplicação Web feita em Python em um Container

Como era resolvido antes

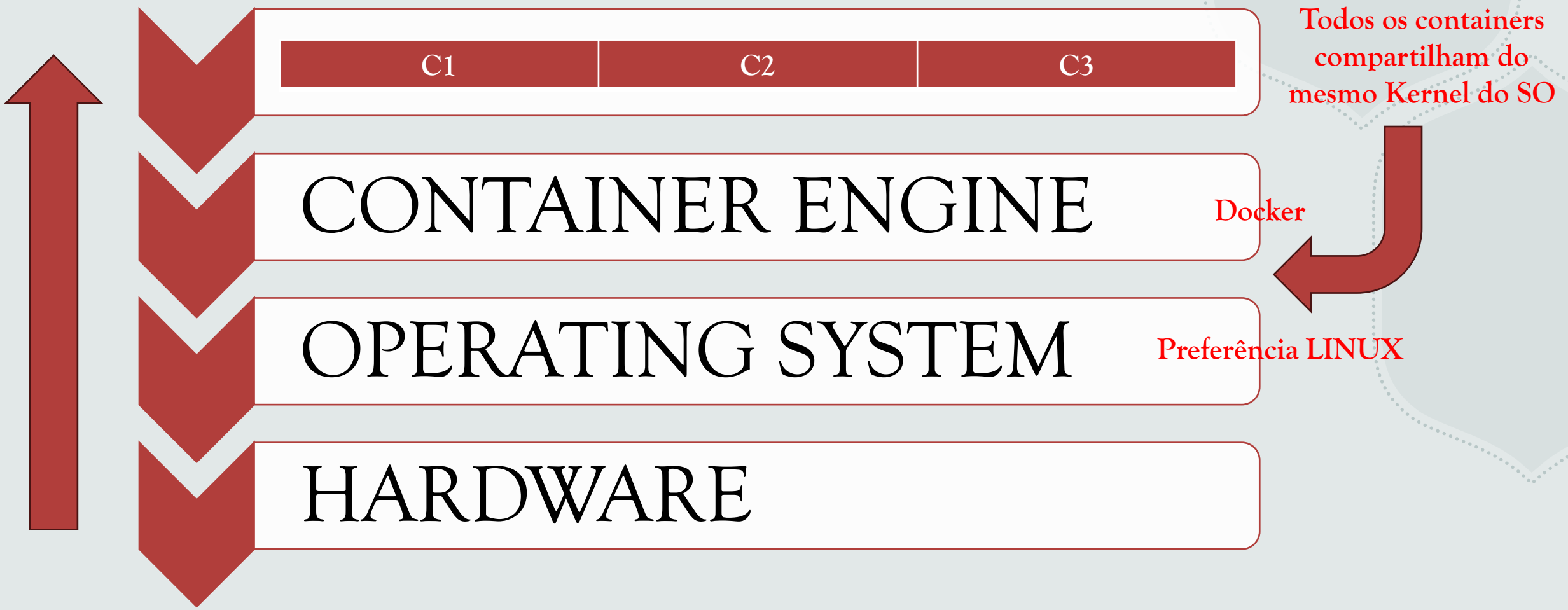


Problemas...

1. Desperdício de Hardware...
2. Difícil gerenciamento de libs
3. Não é imutável

Containers

As aplicações são isoladas: conceito de namespaces



Todos os containers compartilham do mesmo Kernel do SO

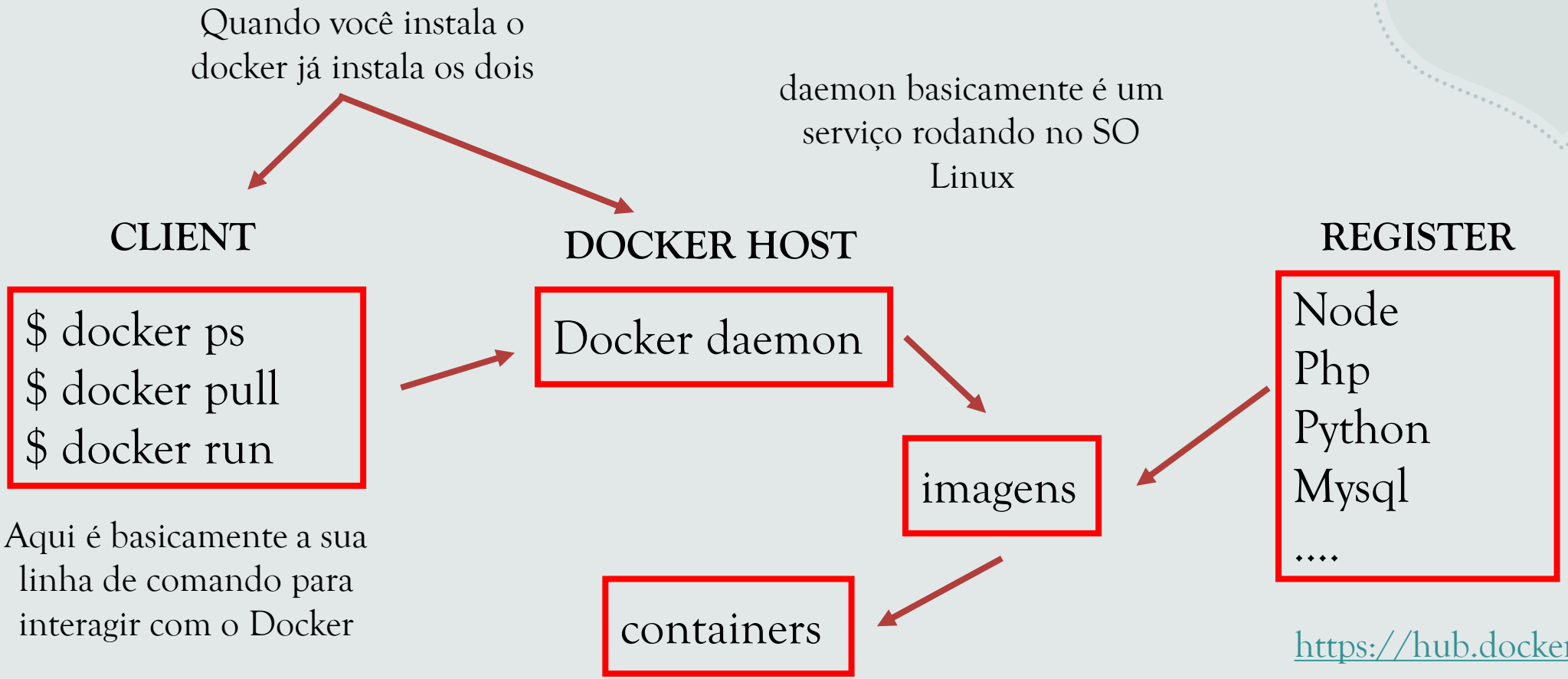
Docker

Preferência LINUX

Vantagens...

1. Otimização de recursos...
2. Empacotamento da aplicação...
3. Imutabilidade...
4. Facilidade no deploy.

Arquitetura do Docker

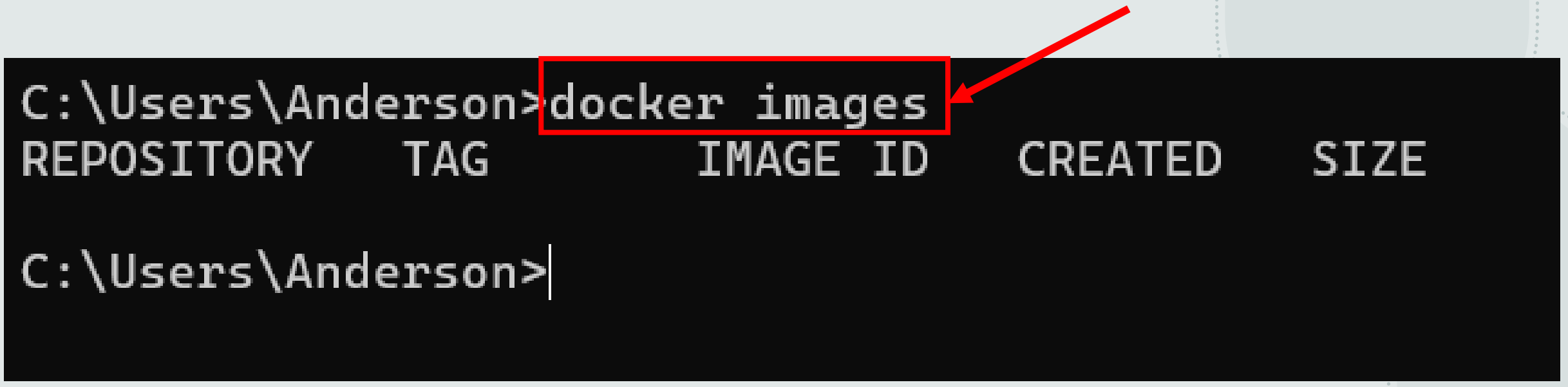


Podemos apontar o Client para vários Docker daemon em máquinas diferentes (outros servidores). DOCKER_HOST

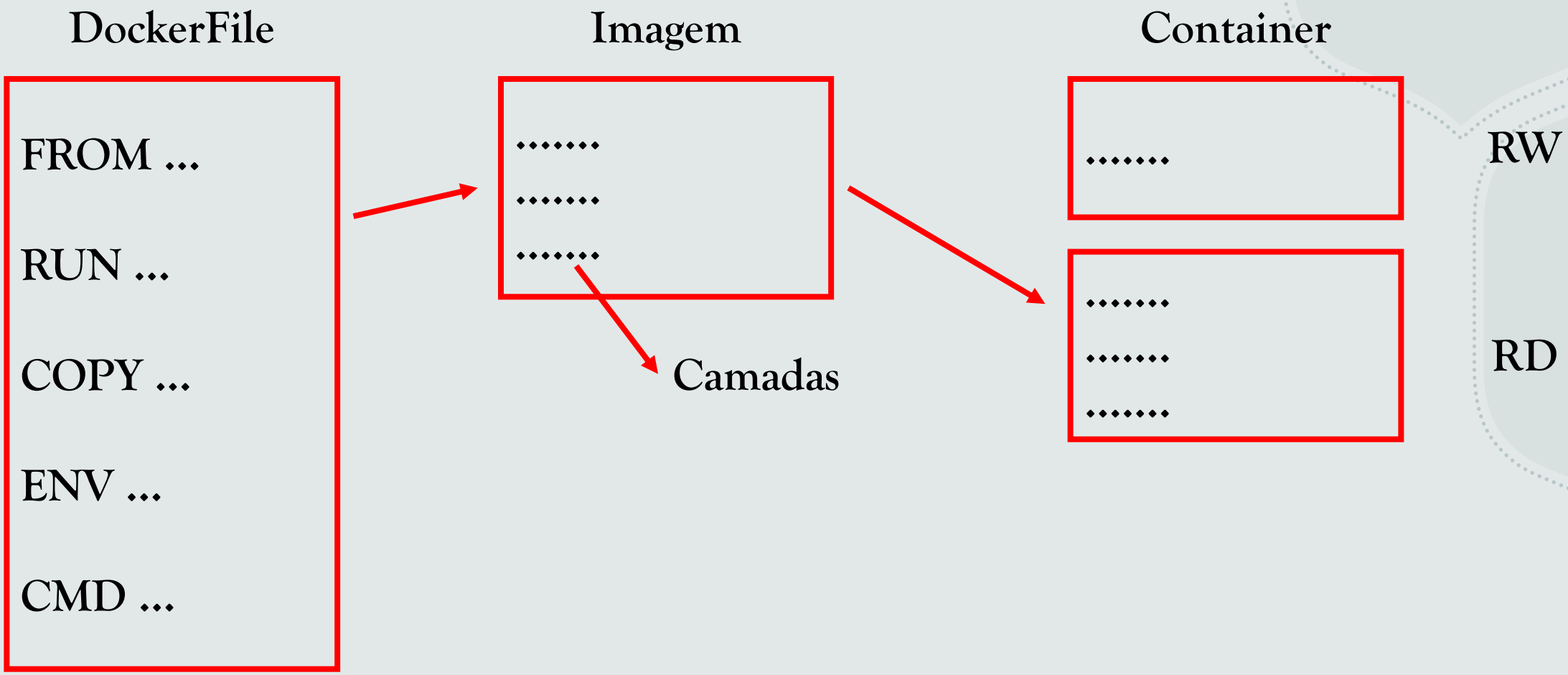
Prática

- Necessário já ter o docker instalado em sua máquina!
- Ter o python instalado
- VS Code

Verificar as imagens no docker



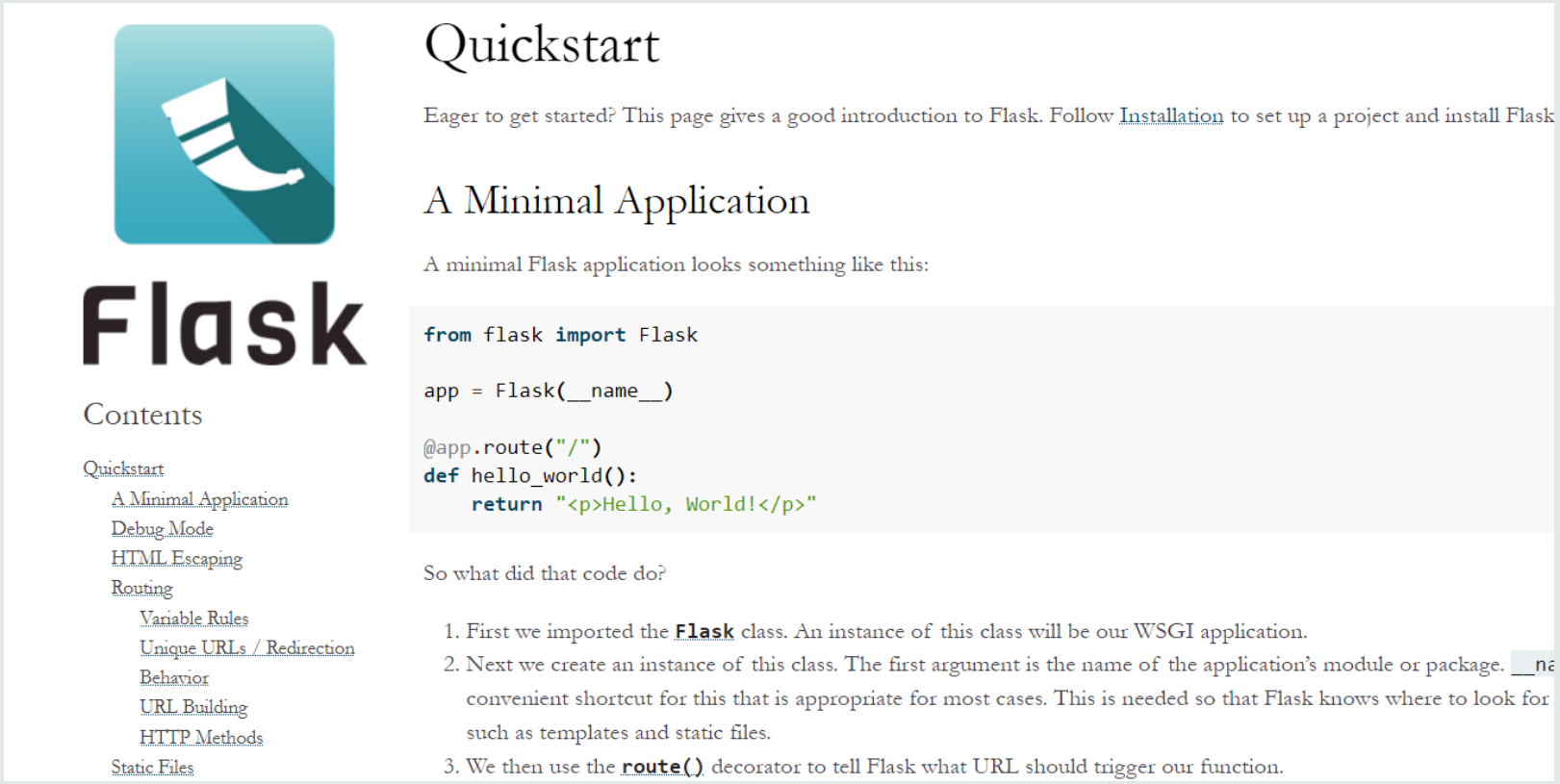
Criar uma imagem docker



Neste arquivo colocamos as instruções para a criação da Imagem

Flask

- Framework para aplicações Web utilizando Python
- https://flask.palletsprojects.com/en/3.0.x/quickstart/#a-minimal-application



Flask

Contents

Quickstart

Eager to get started? This page gives a good introduction to Flask. Follow Installation to set up a project and install Flask

A Minimal Application

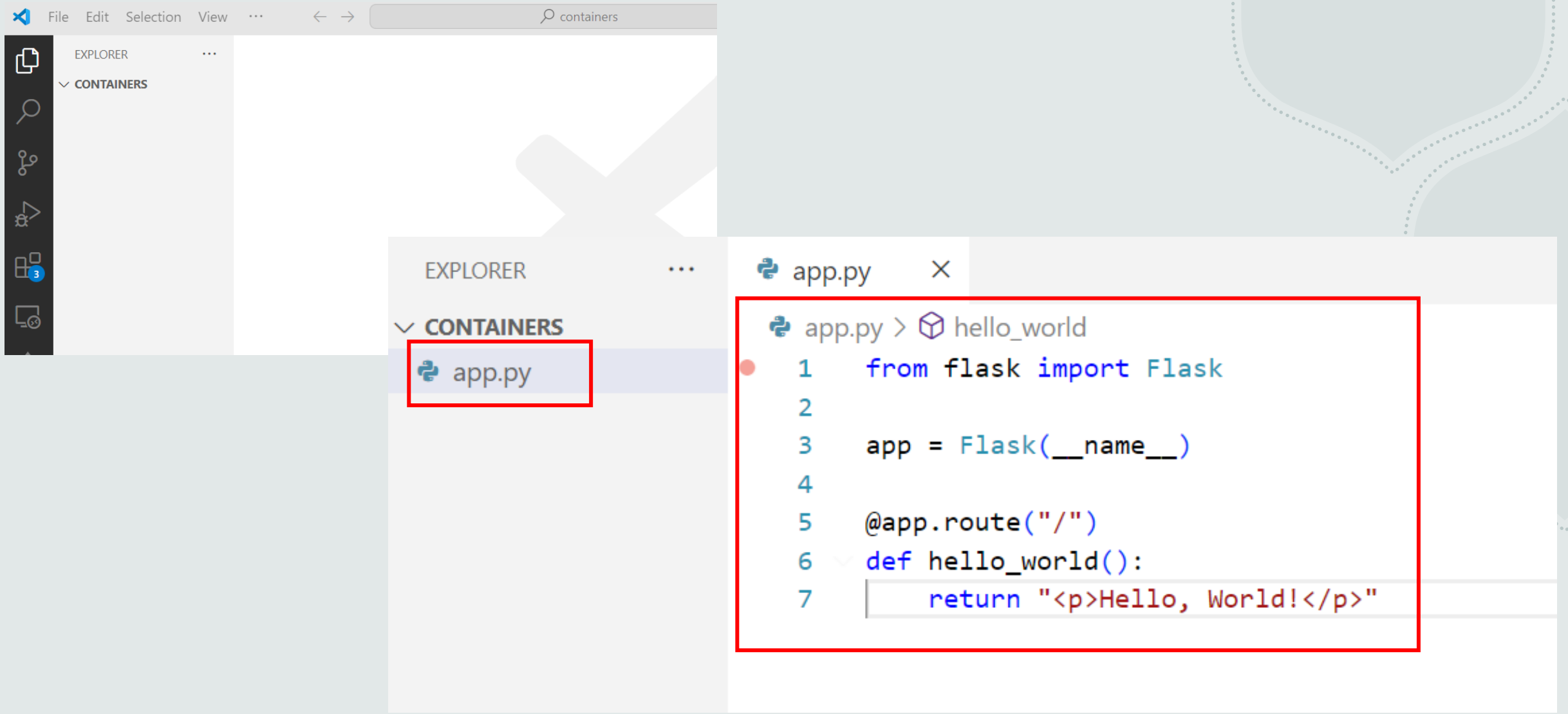
A minimal Flask application looks something like this:

```
from flask import Flask

app = Flask(__name__)

@app.route("/")
def hello_world():
    return "<p>Hello, World!</p>"
```

So what did that code do?

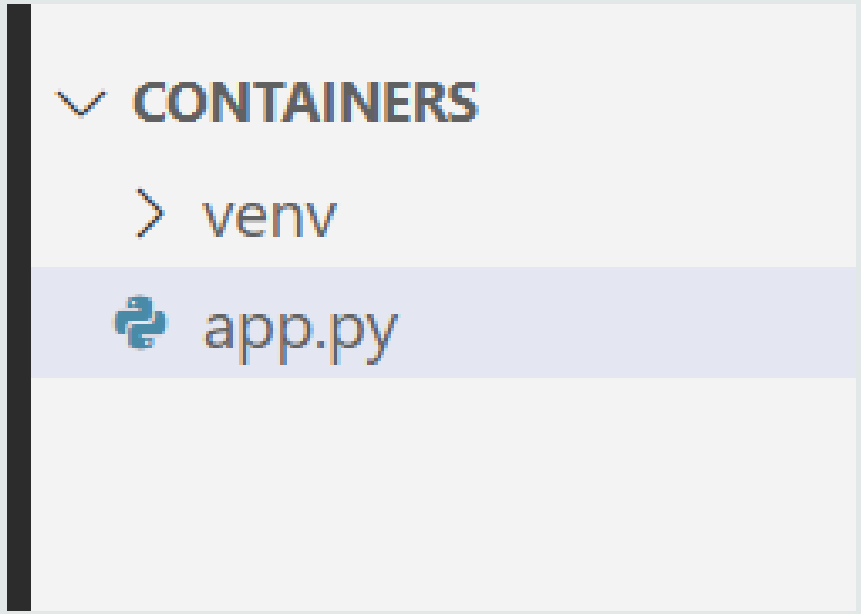
1. First we imported the **Flask** class. An instance of this class will be our WSGI application.
2. Next we create an instance of this class. The first argument is the name of the application's module or package. __na convenient shortcut for this that is appropriate for most cases. This is needed so that Flask knows where to look for such as templates and static files.
3. We then use the **route()** decorator to tell Flask what URL should trigger our function.

Abra uma pasta vazia



Ambiente virtual python

```
PS C:\Users\Anderson\Desktop\containers> python -m venv venv
PS C:\Users\Anderson\Desktop\containers>
```



Ativando o ambiente virtual do python

```
C:\Users\Anderson\Desktop\containers>cd venv/Scripts

C:\Users\Anderson\Desktop\containers\venv\Scripts>Activate

(venv) C:\Users\Anderson\Desktop\containers\venv\Scripts>
```

Volte para a raiz do projeto

```
(venv) C:\Users\Anderson\Desktop\containers\venv\Scripts>cd..

(venv) C:\Users\Anderson\Desktop\containers\venv>cd..

(venv) C:\Users\Anderson\Desktop\containers>
```

Execute o programa app.py

```
(venv) C:\Users\Anderson\Desktop\containers>python app.py
Traceback (most recent call last):
  File "C:\Users\Anderson\Desktop\containers\app.py", line 1, in <module>
    from flask import Flask
ModuleNotFoundError: No module named 'flask'

(venv) C:\Users\Anderson\Desktop\containers>
```

O ambiente virtual do python não tem o módulo Flask instalado, então vamos precisar instalá-lo.

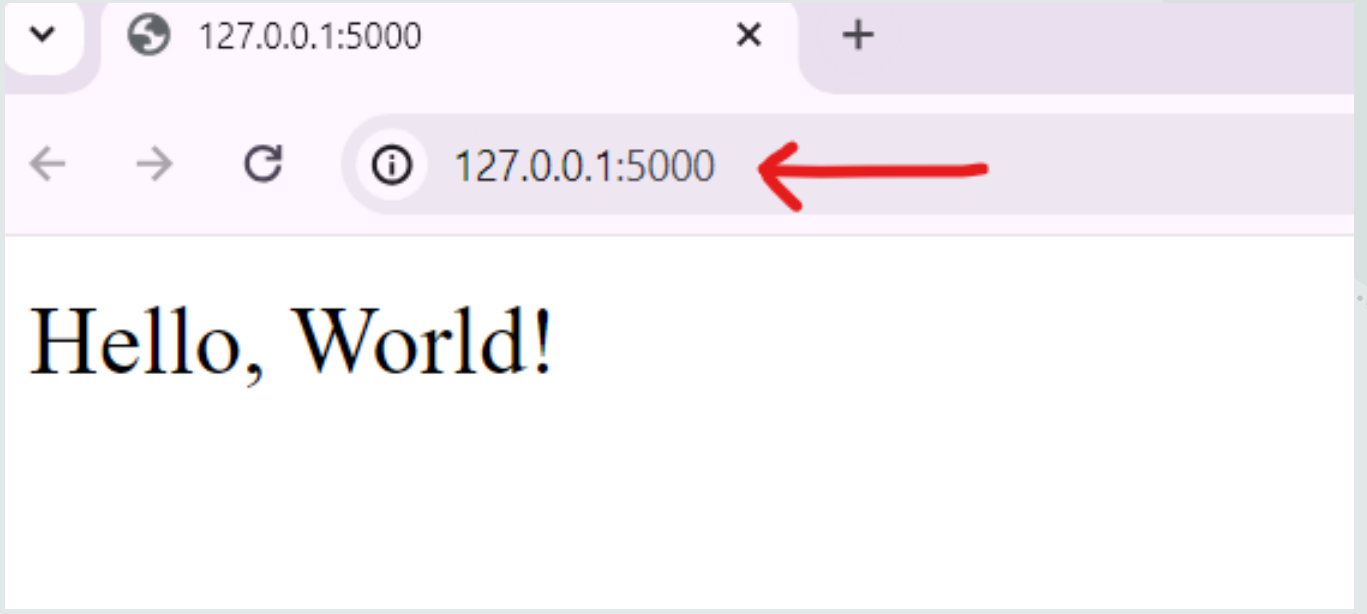
Instalando o Flask

```
(venv) C:\Users\Anderson\Desktop\containers>pip install flask
Collecting flask
  Obtaining dependency information for flask from https://files.pythonhosted.org/pa
e1da13ad9300f87c93af113edd0638c75138c42a0994becfacac078c06/flask-3.0.3-py3-none-any
  Downloading flask-3.0.3-py3-none-any.whl.metadata (3.2 kB)
Collecting Werkzeug>=3.0.0 (from flask)
  Obtaining dependency information for Werkzeug>=3.0.0 from https://files.pythonhos
s/9d/6e/e792999e816d19d7fcbfa94c730936750036d65656a76a5a688b57a656c4/werkzeug-3.0.3
hl.metadata
```

```
Successfully installed Jinja2-3.1.4 MarkupSafe-2.1.5 Werkzeug-3.0.3 blinker-1
rama-0.4.6 flask-3.0.3 itsdangerous-2.2.0

[notice] A new release of pip is available: 23.2.1 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip

(venv) C:\Users\Anderson\Desktop\containers>
```

Rodando a aplicação localmente

```
(venv) C:\Users\Anderson\Desktop\containers>flask run
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deploym
WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

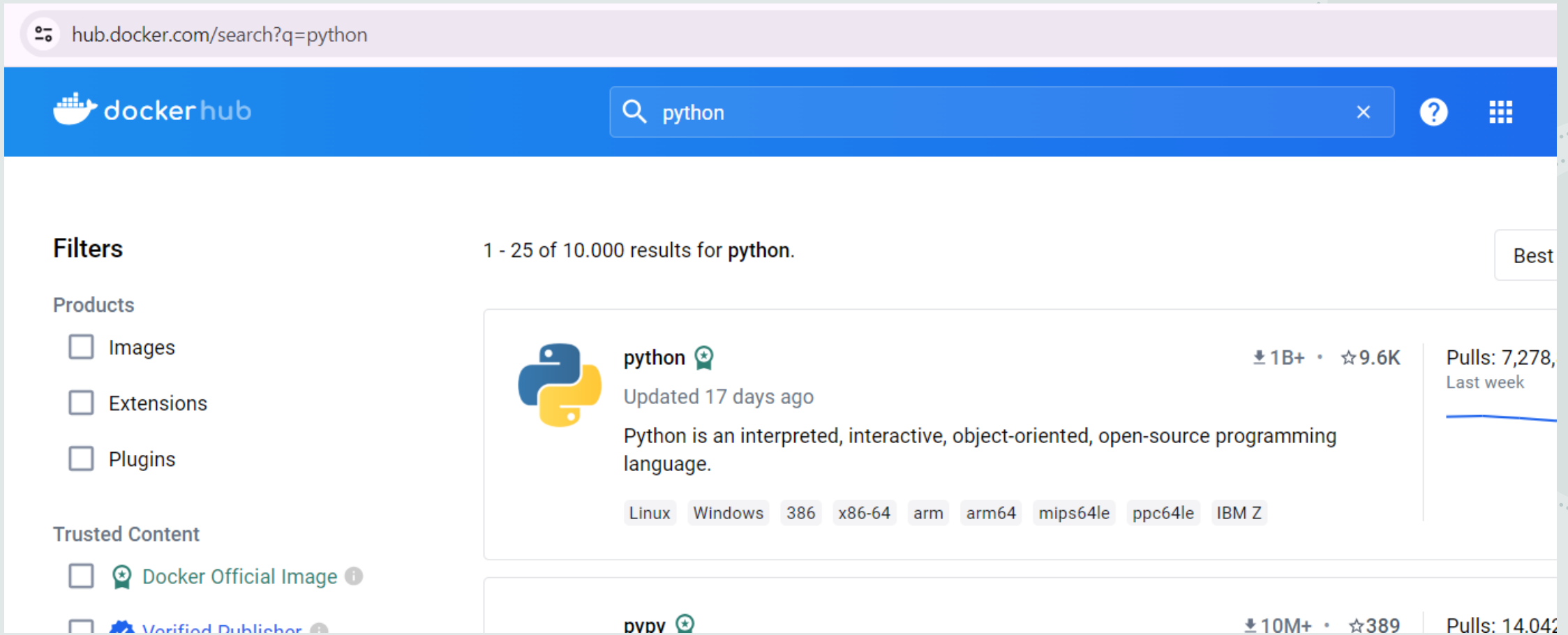


Dockerfile

Agora temos uma aplicação rodando localmente. Agora precisamos criar um arquivo Dockerfile que será o responsável pela criação da Imagem e posterior containers gerados.

https://docs.docker.com/reference/dockerfile/

Docker hub


hub.docker.com/search?q=python
dockerhub
python
Filters
1 - 25 of 10.000 results for python.
Best
Products
Images
Extensions
Plugins
Trusted Content
Docker Official Image
python
1B+ • ☆9.6K
Updated 17 days ago
Python is an interpreted, interactive, object-oriented, open-source programming language.
Linux Windows 386 x86-64 arm arm64 mips64le ppc64le IBM Z
Pulls: 7,278,
Last week
pypy
10M+ • ☆389
Pulls: 14.042

Docker hub

Simple Tags

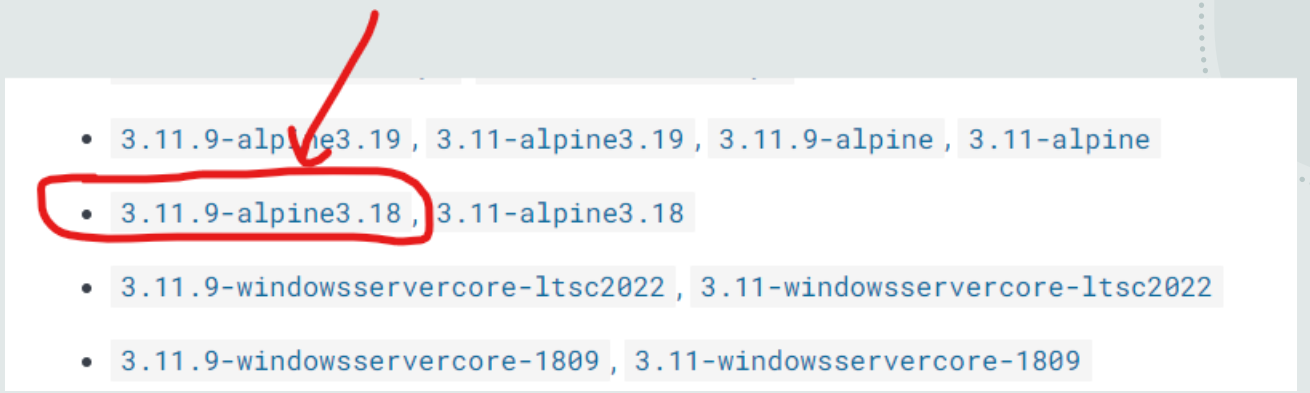
- `3.13.0b1-bookworm` , `3.13-rc-bookworm`
- `3.13.0b1-slim-bookworm` , `3.13-rc-slim-bookworm` , `3.13.0b1-slim` , `3.13-rc-slim`
- `3.13.0b1-bullseye` , `3.13-rc-bullseye`
- `3.13.0b1-slim-bullseye` , `3.13-rc-slim-bullseye`
- `3.13.0b1-alpine3.19` , `3.13-rc-alpine3.19` , `3.13.0b1-alpine` , `3.13-rc-alpine`
- `3.13.0b1-alpine3.18` , `3.13-rc-alpine3.18`
- `3.13.0b1-windowsservercore-ltsc2022` , `3.13-rc-windowsservercore-ltsc2022`
- `3.13.0b1-windowsservercore-1809` , `3.13-rc-windowsservercore-1809`
- `3.12.3-bookworm` , `3.12-bookworm` , `3-bookworm` , `bookworm`
- `3.12.3-slim-bookworm` , `3.12-slim-bookworm` , `3-slim-bookworm` , `slim-bookworm` , `3.12.3-sl`
  `3.12-slim` , `3-slim` , `slim`
- `3.12.3-bullseye` , `3.12-bullseye` , `3-bullseye` , `bullseye`
- `3.12.3-slim-bullseye` , `3.12-slim-bullseye` , `3-slim-bullseye` , `slim-bullseye`
- `3.12.3-alpine3.19` , `3.12-alpine3.19` , `3-alpine3.19` , `alpine3.19` , `3.12.3-alpine` , `3.12-alpine` , `3-alpine` , `alpine`
- `3.12.3-alpine3.18` , `3.12-alpine3.18` , `3-alpine3.18` , `alpine3.18`
- `3.12.3-windowsservercore-ltsc2022` , `3.12-windowsservercore-ltsc2022` , `3-windowsservercore-ltsc2022` , `windowsservercore-ltsc2022`

Repare que existem várias versões para o python. Lembre-se que já temos uma aplicação local rodando em alguma versão já instalada do python!

Verificando a sua versão do python (local)

```
(venv) C:\Users\Anderson\Desktop\containers>python --version
Python 3.11.5

(venv) C:\Users\Anderson\Desktop\containers>
```

Lá no docker hub vamos selecionar uma versão próxima do python para a criação da nossa imagem do python



Baixando a versão do python

```
(venv) C:\Users\Anderson\Desktop\containers>docker pull python:3.11.9-alpine3.18
```

```
(venv) C:\Users\Anderson\Desktop\containers>docker pull python:3.11.9-alpine3.18
3.11.9-alpine3.18: Pulling from library/python
619be1103602: Pull complete
36988be9c68b: Pull complete
46d5b401bdc2: Downloading  6.974MB/12.5MB
4e0054417e48: Download complete
47f039281c77: Download complete
```

docker pull python:3.11.9-alpine3.18

```
Status: Downloaded newer image for python:3.11.9-alpine3.18
docker.io/library/python:3.11.9-alpine3.18

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview python:3.11
.9-alpine3.18

(venv) C:\Users\Anderson\Desktop\containers>
```

Verificando a imagem baixada

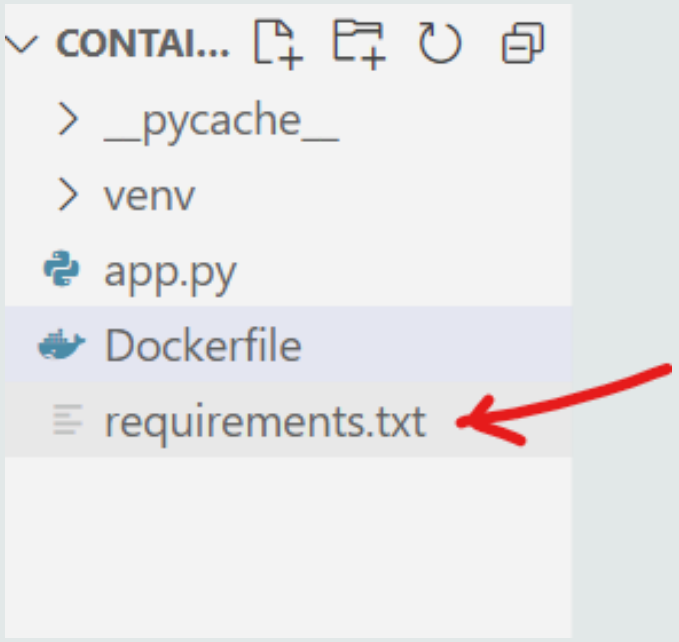
```
(venv) C:\Users\Anderson\Desktop\containers>docker images
REPOSITORY   TAG                 IMAGE ID       CREATED        SIZE
python       3.11.9-alpine3.18   f78601fd3a0f   5 weeks ago    52.3MB

(venv) C:\Users\Anderson\Desktop\containers>
```

docker images

Gerando as dependências de sua aplicação python

```
(venv) C:\Users\Anderson\Desktop\containers>pip freeze > requirements.txt
```

pip freeze > requirements.txt



requirements.txt

```
1   blinker==1.8.2
2   click==8.1.7
3   colorama==0.4.6
4   Flask==3.0.3
5   itsdangerous==2.2.0
6   Jinja2==3.1.4
7   MarkupSafe==2.1.5
8   Werkzeug==3.0.3
9
```

flask run –host 0.0.0.0

```
(venv) C:\Users\Anderson\Desktop\containers>flask run --host 0.0.0.0
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. U
WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://192.168.0.14:5000
Press CTRL+C to quit
```

Editando o arquivo Dockerfile

Dockerfile > ...

```
1   FROM python:3.11.9-alpine3.18
2
3   WORKDIR /app
4
5   COPY requirements.txt .
6
7   RUN pip install -r requirements.txt
8
9   EXPOSE 5000
10
11  ENV FLASK_ENV development
12
13  COPY app.py .
14
15  CMD [ "flask", "run", "--host", "0.0.0.0" ]
```

Criando a imagem

```
docker build -t flask-app:1.0.0 .
```

```
(venv) C:\Users\Anderson\Desktop\containers>docker build -t flask-app:1.0.0 .
[+] Building 26.8s (10/10) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 258B
 => [internal] load metadata for docker.io/library/python:3.11.9-alpine3.18
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [1/5] FROM docker.io/library/python:3.11.9-alpine3.18
 => [internal] load build context
 => => transferring context: 335B
 => [2/5] WORKDIR /app
 => [3/5] COPY requirements.txt .
 => [4/5] RUN pip install -r requirements.txt
 => [5/5] COPY app.py .
 => exporting to image
 => => exporting layers
 => => writing image sha256:1fbea5b4e64e101915082de848fb14977b89d51bed1bd28d6c3
 => => naming to docker.io/library/flask-app:1.0.0
```

Verificando as imagens criadas

```
(venv) C:\Users\Anderson\Desktop\containers>docker images
REPOSITORY   TAG                 IMAGE ID       CREATED              SIZE
flask-app    1.0.0               1fbea5b4e64e   About a minute ago   68.8MB
python       3.11.9-alpine3.18   f78601fd3a0f   5 weeks ago          52.3MB
```

Verificando os containers que estão rodando

`docker ps`

```
(venv) C:\Users\Anderson\Desktop\containers>docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES

(venv) C:\Users\Anderson\Desktop\containers>
```

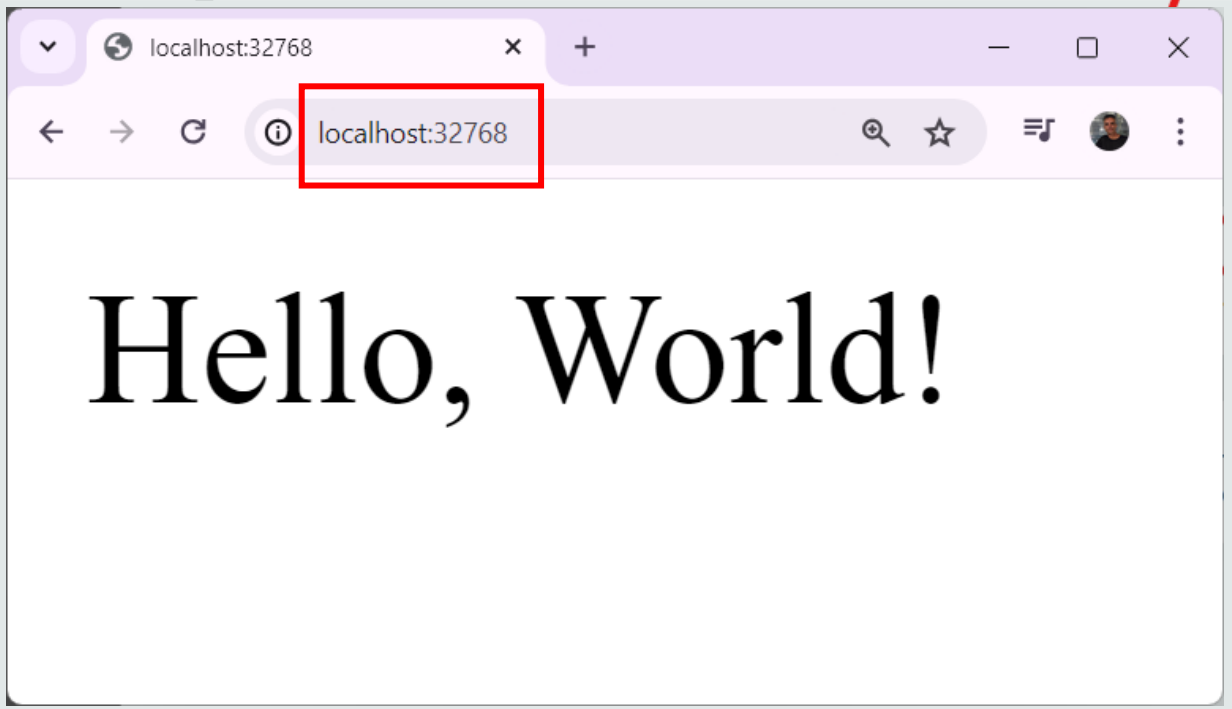
Por enquanto ainda não temos nenhum container rodando!

Criando um container

docker run -d -P flask-app:1.0.0

```
(venv) C:\Users\Anderson\Desktop\containers>docker run -d -P flask-app:1.0.0
d223f432d90c547f8c4fc8dd44ad17fb20e3938b4a86eb93f3c0dc0362a2e9ae
```

```
(venv) C:\Users\Anderson\Desktop\containers>docker ps
CONTAINER ID   IMAGE             COMMAND                  CREATED          STATUS          PORTS                     NAMES
d223f432d90c   flask-app:1.0.0   "flask run --host 0.…"   43 seconds ago   Up 42 seconds   0.0.0.0:32768->5000/tcp   busy_shamir
```



Se você executar novamente o docker run, ele irá criado um outro container

```
(venv) C:\Users\Anderson\Desktop\containers>docker run -d -P flask-app:1.0.0
df79127b71cab858c5194b76bbe5be34abd188fe5a04b3f9da672c1a7f93d0b3

(venv) C:\Users\Anderson\Desktop\containers>docker ps
CONTAINER ID   IMAGE             COMMAND                  CREATED         STATUS         PORTS                     NAMES
df79127b71ca   flask-app:1.0.0   "flask run --host 0.…"   4 seconds ago   Up 3 seconds   0.0.0.0:32769->5000/tcp   goofy_margulis
d223f432d90c   flask-app:1.0.0   "flask run --host 0.…"   4 minutes ago   Up 4 minutes   0.0.0.0:32768->5000/tcp   busy_shamir
```

Verificando os logs dos containers

```
(venv) C:\Users\Anderson\Desktop\containers>docker ps
CONTAINER ID   IMAGE              COMMAND                  CREATED          STATUS
df79127b71ca   flask-app:1.0.0    "flask run --host 0.…"   4 seconds ago    Up 3 second
d223f432d90c   flask-app:1.0.0    "flask run --host 0.…"   4 minutes ago    Up 4 minute

(venv) C:\Users\Anderson\Desktop\containers>docker logs df79127b71ca
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://172.17.0.3:5000
Press CTRL+C to quit
```

docker logs xxxxxxxxxxx

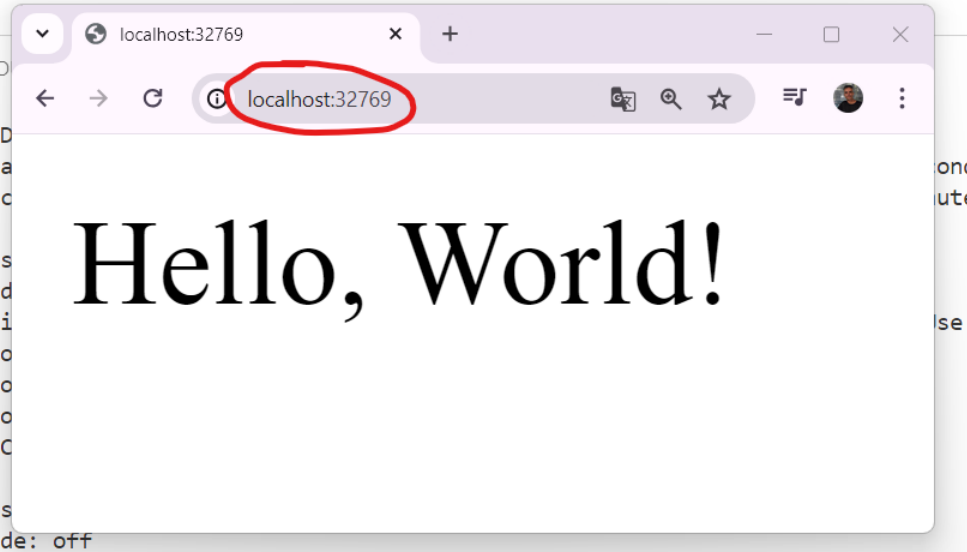
Aqui é o id do container que você quer verificar os logs

Verificando os logs dos containers em tempo real

```
(venv) C:\Users\Anderson\Desktop\containers>docker logs df79127b71ca -f
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a product
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://172.17.0.3:5000
Press CTRL+C to quit
```

docker logs xxxxxxxxxxx -f

```
1   FROM python:3.11.9-alpine3.18
2
```




```
PROBLEMS
CONTAINER ID                                                    PORTS
df79127b71ca                                                    0.0.0.0:327
d223f432d90c                                                    0.0.0.0:327

(venv) C:\Us
 * Debug mod
WARNING: Thi                                                   se a production WS
 * Running o
 * Running o
 * Running o
Press CTRL+C

(venv) C:\Us
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WS
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://172.17.0.3:5000
Press CTRL+C to quit
172.17.0.1 - - [11/May/2024 21:02:30] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [11/May/2024 21:02:30] "GET /favicon.ico HTTP/1.1" 404 -
```

Parando a execução de um container

```
(venv) C:\Users\Anderson\Desktop\containers>docker ps
CONTAINER ID   IMAGE             COMMAND                  CREATED          STATUS          PORTS                     NAMES
df79127b71ca   flask-app:1.0.0   "flask run --host 0.…"   19 minutes ago   Up 19 minutes   0.0.0.0:32769->5000/tcp   goofy_margulis
d223f432d90c   flask-app:1.0.0   "flask run --host 0.…"   23 minutes ago   Up 23 minutes   0.0.0.0:32768->5000/tcp   busy_shamir

(venv) C:\Users\Anderson\Desktop\containers>docker stop df79127b71ca
df79127b71ca

(venv) C:\Users\Anderson\Desktop\containers>docker ps
CONTAINER ID   IMAGE             COMMAND                  CREATED          STATUS          PORTS                     NAMES
d223f432d90c   flask-app:1.0.0   "flask run --host 0.…"   24 minutes ago   Up 24 minutes   0.0.0.0:32768->5000/tcp   busy_shamir

(venv) C:\Users\Anderson\Desktop\containers>docker stop d223f432d90c
d223f432d90c

(venv) C:\Users\Anderson\Desktop\containers>docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES

(venv) C:\Users\Anderson\Desktop\containers>
```

docker stop xxxxxxxxxxx