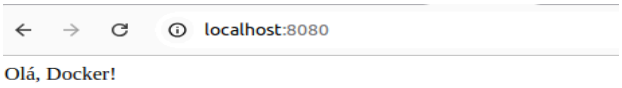
Universidade Senac Tecnologia
Pelotas - Rio Grande Do Sul

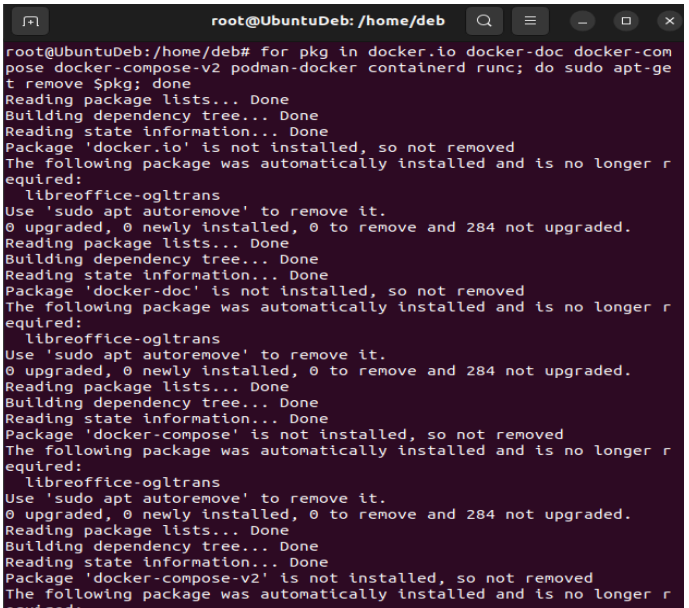
Computação em Nuvem - Docker

Disciplina de Computação em Nuvem no semestre de 2024/2

Por Débora Sieburger Carvalho



Primeiramente, após adentrar o sistema operacional "*linux ubuntu*" , instalei o Google Chrome e comecei a instalar o ***Docker*** através da documentação descrita no site oficial. Código de instalação:



for pkg in docker.io docker-doc docker-compose docker-compose-v2 podman-docker containerd runc; do sudo apt-get remove $pkg; done

Dessa forma a gente tira arquivos conflitantes com a instalação do docker.

Instalando usando o apt repositório:
● sudo apt-get update
● sudo apt-get install ca-certificates curl
● sudo install -m 0755 -d /etc/apt/keyrings
● sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
● sudo chmod a+r /etc/apt/keyrings/docker.asc

Instalando usando o apt repositório:

● sudo apt-get update

● sudo apt-get install ca-certificates curl

● sudo install -m 0755 -d /etc/apt/keyrings

● sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc

● sudo chmod a+r /etc/apt/keyrings/docker.asc

Adicionando ao repositório: ● echo \ "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \ $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \

● sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

● sudo apt-get update

Para instalar a versão mais recente, executei ● sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

Verificando se a instalação do Docker Engine foi bem-sucedida executando a hello-world imagem: ● sudo service docker start

● sudo docker run hello-world

● sudo service docker status

● sudo service docker start

● sudo service docker stop

Inicializando quando iniciar a máquina: ○ sudo systemctl enable docker.service

○ sudo systemctl enable containerd.service

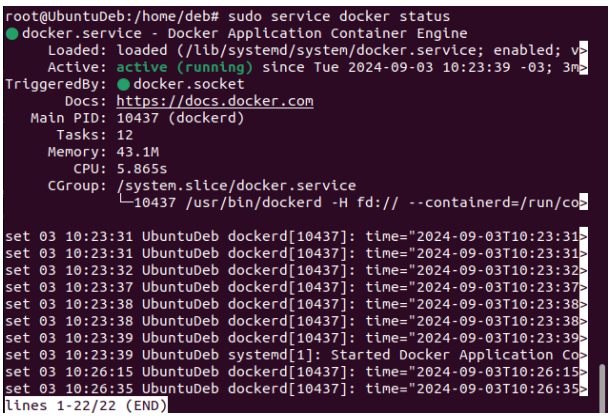
Parar a inicialização: ○ sudo systemctl disable docker.service

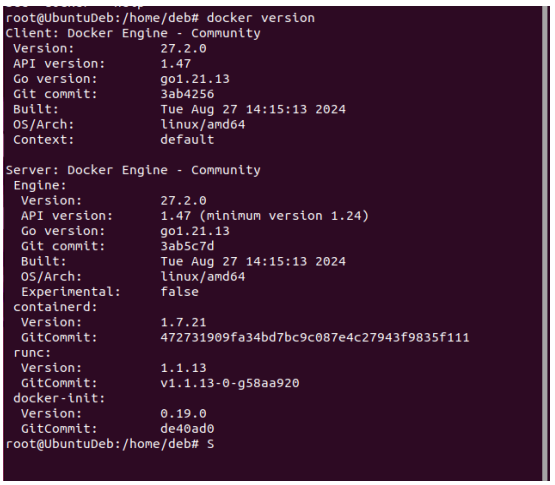
○ sudo systemctl disable containerd.service

● docker version

Verificando os docker ativos: ● docker ps

docker container ls -a



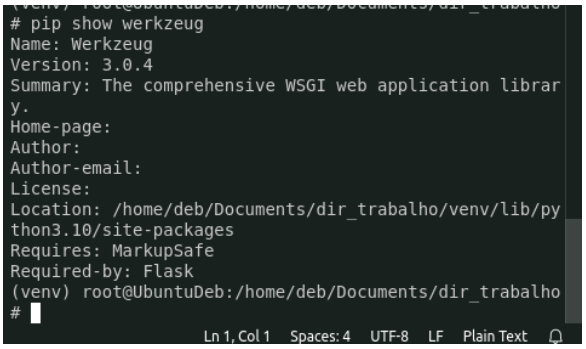


Criei um diretório e comecei a instalação do Flask.

Baixei o python pelo diretorio root:

sudo **apt install python3-venv**

Criei um dockerfie e um requirements.txt



python3 -m venv venv
sudo apt install python3.10-venv source venv/bin/activate
Instalar dependências no python:
pip3 install --no-cache-dir -r requirements.txt
pip install --upgrade flask
python app.py pip freeze > requerements.txt
pip show werkzeug
deactivate

Após isso criei a imagem docker a partir do código:
docker build -t minha-imagem:v1 .

- **Dockerfile**

Dockerfile

```
FROM ubuntu:22.04
RUN apt-get update && \
apt-get install python3.11 python3.11-dev python3-pip -y
WORKDIR /app
COPY . .
RUN pip3 install --no-cache-dir -r requirements.txt
EXPOSE 8080
ENV LOGOMARCA="iamgem(link)"
ENV FOTO="iamgem(link)"
ENV NOME="DEB"
ENV IDADE="DEB"
ENV EMAIL="DEB"
ENV PROFISSAO="DEB"
ENV SITE="DEB"
CMD [ "python3","app.py" ]
```

Criei um arquivo requirements.txt e outro app.py

python3 -m venv venv

- **sudo apt install python3.10-venv source venv/bin/activate**
- **pip3 install --no-cache-dir -r requirements.txt**
- **pip install --upgrade flask**
- **python app.py pip freeze > requerements.txt**
- **pip show werkzeug deactivate**

- **Image dir**

  - **docker run nome-da-imagem**

docker build -t minha-imagem:v1 . (o comando tem o ponto (.) de pasta local)
docker images docker run -it -p 8080:8080 minha-imagem:v1

docker run -it -p 8080:8080 image_dir:v1