

**Projeto Docker | Servidor WEB**

**Disponibilizar o clean-architecture-calculator em um servidor web utilizando o Docker.**

Prof. MSc. Emmanoel Monteiro S. Junior

**Passo 1: Verificar a Imagem**

Antes de criar o contêiner, você pode verificar se a imagem já está presente localmente ou se precisa ser baixada do Docker Hub.

Para listar as imagens disponíveis em sua máquina, use o comando:

**docker** images

*OBS: Se a imagem ubuntu-nginx não estiver na lista, o Docker irá baixá-la automaticamente quando você tentar criar o contêiner.*

**Passo 2: Criar e Executar o Contêiner**

O comando principal para criar e executar um contêiner é **docker run**. Ele possui várias opções que você pode usar para configurar o comportamento do contêiner.

Use o seguinte comando no seu terminal:

**docker run** -d -p 8080:80 --name **meu-servidor-web** ubuntu/nginx:latest

**Passo 3: Verificar o Contêiner em Execução**

Para garantir que o contêiner foi criado e está rodando corretamente, você pode usar o comando **docker ps**.

**docker** ps

*OBS: Este comando lista todos os contêineres que estão em execução. Você deverá ver uma linha com o nome meu-servidor-web e o status Up (ativo).*

**Passo 4: Acessar o Servidor Web**

Com o contêiner em execução, você pode abrir seu navegador e acessar o endereço http://localhost:8080.

Você verá a página de boas-vindas padrão do Nginx, que é o conteúdo servido por essa imagem por padrão. Isso confirma que o contêiner está funcionando e que o mapeamento de portas foi bem-sucedido.

**Passo 5: Copiar arquivos do projeto local para a pasta html dentro do container criado.**

Agora, use o comando docker cp para copiar seus arquivos. A sintaxe é:

docker cp [caminho\_local] [nome\_do\_container]:[caminho\_no\_container]

Assumindo que seus arquivos estão na pasta atual, e você quer copiar tudo para /var/www/html:

**docker** cp .\ calculadora\_completa.html **meu-servidor-web**:/var/www/html

Teste chamando o http://localhost:8080/calculadora\_completa.html . Agora a pagina da calculadora está dentro do serviço web do container

**Passo 6: Mudar a index do serviço web do nginx.**

Agora, vamos copiar o arquivo de configuração do serviço web do nginx para nossa máquina local.

**docker** cp **meu-servidor-web**:/etc/nginx/sites-enabled/default **.\**

Comente a linha

# **index** index.html index.htm index.nginx-debian.html;

Adicione logo a baixo

**index**  calculadora\_completa.html;

Salve e copie o arquivo de volta para o container

**docker** **.\** default cp **meu-servidor-web**:/etc/nginx/sites-enabled/

Reinicie o serviço do nginx

**docker** exec **meu-servidor-web** service nginx restart

Teste chamando o http://localhost:8080 e a calculadora deve aparecer como página principal.

**AQUI ESTÁ UMA TABELA COM OS PRINCIPAIS COMANDOS DOCKER COM A SINTAXE E SEUS RESPECTIVOS OBJETIVOS.**

| **Comando Docker** | **Sintaxe de Comando** | **Objetivo** |
| --- | --- | --- |
| **docker build** | docker build -t [nome\_da\_imagem] . | Constrói uma imagem Docker a partir de um Dockerfile. A flag -t (tag) nomeia a imagem, e o . indica o diretório atual como contexto. |
| **docker run** | docker run -d -p [porta\_host]:[porta\_container] --name [nome\_container] [nome\_da\_imagem] | Cria e inicia um novo contêiner a partir de uma imagem. As flags -d (detached), -p (publish) e --name são usadas para rodar em segundo plano, mapear portas e nomear o contêiner, respectivamente. |
| **docker images** | docker images | Lista todas as imagens Docker que estão armazenadas localmente em sua máquina. |
| **docker ps** | docker ps | Lista todos os contêineres que estão atualmente em execução. |
| **docker ps -a** | docker ps -a | Lista todos os contêineres, incluindo aqueles que foram parados. |
| **docker stop** | docker stop [nome\_do\_container] | Envia um sinal para que um contêiner em execução pare de forma graciosa. |
| **docker start** | docker start [nome\_do\_container] | Inicia um contêiner que foi previamente parado. |
| **docker rm** | docker rm [nome\_do\_container] | Remove um contêiner. O contêiner deve estar parado antes de ser removido. |
| **docker rmi** | docker rmi [nome\_da\_imagem] | Remove uma imagem Docker do seu armazenamento local. |
| **docker logs** | docker logs [nome\_do\_container] | Exibe os logs gerados pelo contêiner, o que é útil para depuração. |
| **docker exec** | docker exec -it [nome\_do\_container] [comando] | Executa um comando dentro de um contêiner em execução. A flag -it permite uma interação com o terminal, como em docker exec -it [nome\_do\_container] /bin/bash. |