

Disciplina: Redes de Computadores II.
Professor: Luiz Carlos Efigenio da Rosa Junior.
Aluno: George Lincon Veloso Cruz.

TADS 22

Atividade Docker – DockerHub – HOST

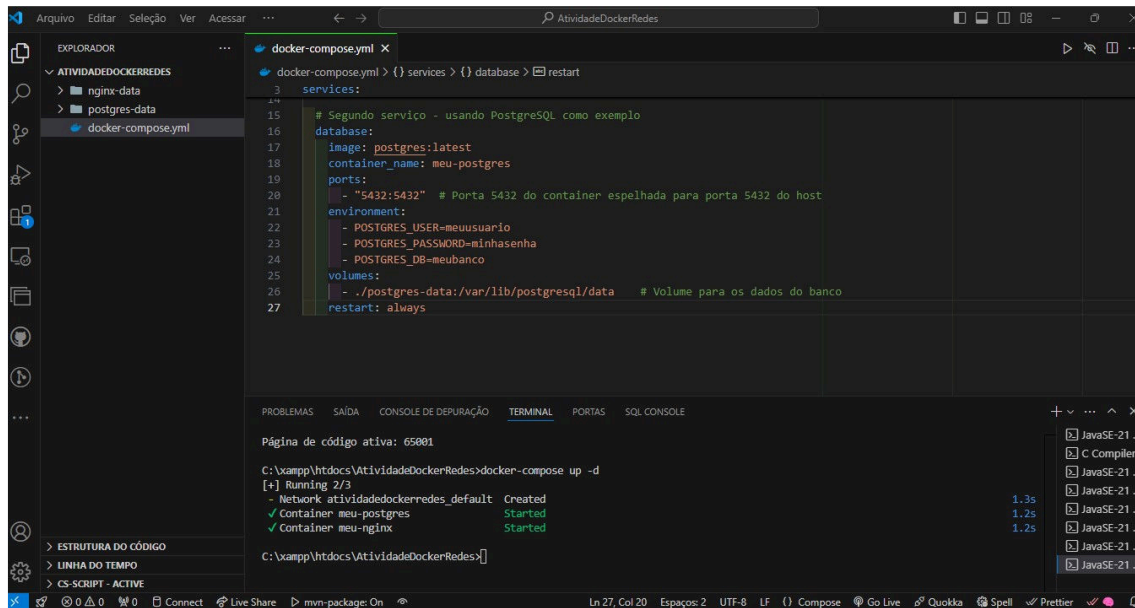
- ✓ Subir dois serviços funcionais a partir da imagem do DockerHub: (espelhados ao o HOST).
- ✓ Explorar os comandos usados com a ferramenta Docker (Documentar no PDF).
- ✓ Entrega: Documentar todo passo em um documento PDF.

Passo a Passo da Atividade:

->>>>> Arquivo docker-compose.yml para subir dois serviços utilizando imagens do Docker Hub, com as portas espelhadas para o host Windows <<<<<-

Para usar este arquivo:

- 1. Abra o VS Code;**
- 2. Crie um Novo Projeto;**
- 3. Novo arquivo: docker-compose.yml;**



(Código dentro da pasta -> novo arquivo -> docker-compose.yml)

version: '3'

services:

Primeiro serviço - usando Nginx como exemplo

webserver:

image: nginx:latest

container_name: meu-nginx

ports:

- "80:80" # Porta 80 do container espelhada para porta 80 do host

- "443:443" # Porta 443 do container espelhada para porta 443 do host

volumes:

- ./nginx-data:/usr/share/nginx/html # Volume para os arquivos do site

restart: always

Segundo serviço - usando PostgreSQL como exemplo

database:

image: postgres:latest

container_name: meu-postgres

ports:

- "5432:5432" # Porta 5432 do container espelhada para porta 5432 do host

environment:

- POSTGRES_USER=meuusuario

- POSTGRES_PASSWORD=minhasenha

- POSTGRES_DB=meubanco

volumes:

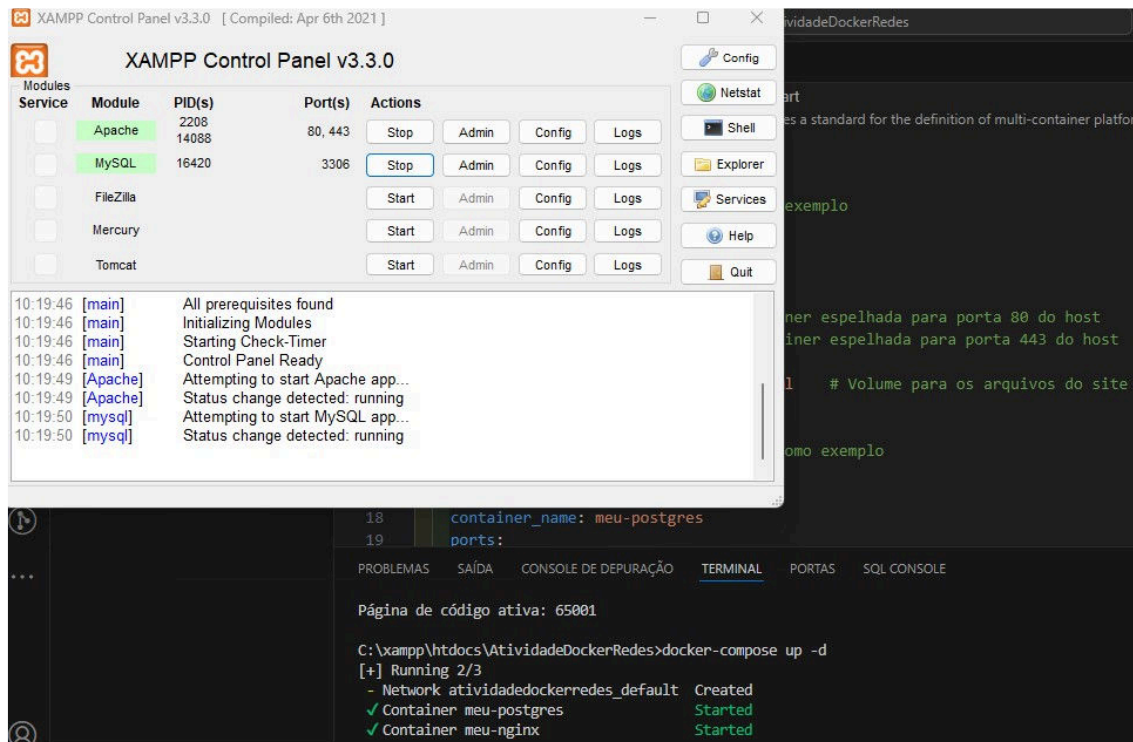
- ./postgres-data:/var/lib/postgresql/data # Volume para os dados do banco

restart: always

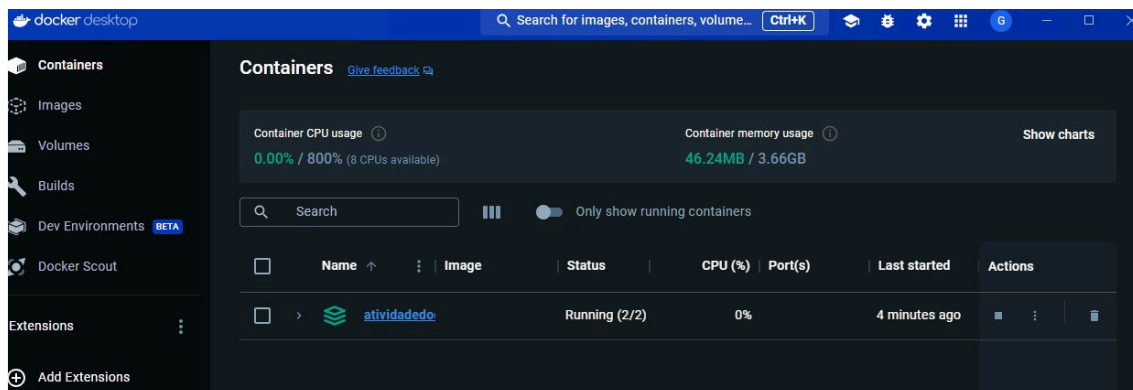
4. Dentro do VS Code – Novo terminal;
5. Execute o comando: `docker-compose up -d`.

Observação: XAMPP e Docker Desktop ativados e rodando o projeto:

XAMPP:



Docker Desktop:



Comandos Básicos do Docker:

- Verificar a versão do Docker: `docker --version`;
- Executar um contêiner: `docker run [opções] imagem [comando]`;
- Listar contêineres em execução: `docker ps`;
- Listar todos os contêineres (incluindo parados): `docker ps -a`;
- Construir uma imagem a partir de um Dockerfile: `docker build -t nome-da-imagem .`;
- Parar um contêiner: `docker stop nome-do-contêiner`;
- Remover um contêiner: `docker rm nome-do-contêiner`;
- Remover uma imagem: `docker rmi nome-da-imagem`;
- Executar um comando em um contêiner em execução: `docker exec -it nome-do-contêiner bash`;
- Ver logs de um contêiner: `docker logs nome-do-contêiner`.

✓ **BÔNUS: Criar uma rede entre elas / E um dos serviços criar o Dockerfile.**

->>>>> Principais mudanças e características <<<<<-

Rede Docker:

Configuração:

1. Estrutura dos diretórios:

```
seu-projeto/  
├── docker-compose.yml  
├── Dockerfile  
├── nginx.conf  
├── site-content/  
│   └── index.html
```

2. Crie um arquivo index.html simples na pasta site-content:

```
<!DOCTYPE html>  
<html>  
<head>  
  <title>Meu Site</title>  
</head>  
<body>  
  <h1>Bem-vindo ao meu site!</h1>  
</body>  
</html>
```

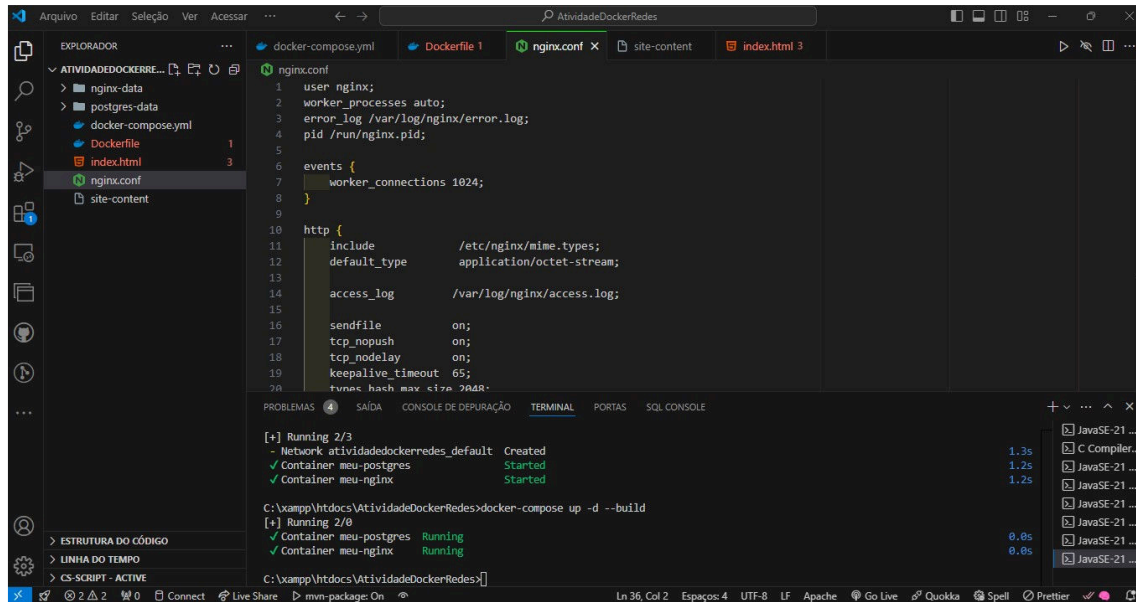
3. Execute os containers:

```
docker-compose up -d --build
```

Características:

Rede Docker:

- Criada uma rede bridge chamada **minha-rede**;
- Ambos os serviços estão conectados a esta rede;
- Permite comunicação entre containers usando seus nomes como hostnames.



(Execução no VS Code)