ROTEIRO PRÁTICO 02

Criação de um contêiner com Docker com Git e Node.js

Objetivo Geral:

O objetivo desta prática é desenvolver habilidades em contêinerização e familiarização com ferramentas DevOps, através da criação de um contêiner Docker contendo Git e Node.js, visando a automação e simplificação do setup de ambientes de desenvolvimento.

Objetivos Específicos:

Familiarizar-se com o Docker: Aprender os conceitos básicos de contêineres, imagens e Dockerfiles.

Praticar o uso do Git: Entender como utilizar o Git dentro de um contêiner para controle de versões.

Explorar o Node.js: Configurar um ambiente Node.js dentro de um contêiner para desenvolvimento de aplicações JavaScript.

Passo a Passo:

1. Preparação do Ambiente:

Instale o Docker em sua máquina, seguindo as instruções oficiais disponíveis na documentação do Docker para o seu sistema operacional.

2. Criação do Dockerfile:

Crie um arquivo na raiz do seu projeto chamado Dockerfile. Este arquivo será utilizado para definir a imagem Docker que contém Git e Node.js.

```
# Utiliza a imagem oficial do Node.js como base
FROM node:latest

# Instala o Git
RUN apt-get update && \
    apt-get install -y git

# Define o diretório de trabalho
WORKDIR /app

# Expõe a porta 3000 para a aplicação Node.js
EXPOSE 3000

# Mantém o contêiner em execução
CMD ["tail", "-f", "/dev/null"]
```

3. Construção da Imagem Docker:

Com o Dockerfile criado, construa a imagem Docker utilizando o comando abaixo.

```
docker build -t dev-tools .
```

4. Execução do Contêiner:

Após a construção da imagem, crie e inicie um contêiner a partir da imagem construída com o seguinte comando:

```
docker run -it --rm -p 8080:8080 dev-tools
```

5. Acesso ao Contêiner e Teste das Ferramentas:

Acesso ao Contêiner:

Caso ainda não tenha associado uma chave SSH ao seu repositório, será necessário criar uma:

No terminal EXEC do Docker, digite o seguinte comando:

```
ssh-keygen -t ed25519 -C "your_email@example.com"
```

Obtenha a chave gerada no caminho root/.ssh/id_ed25519.pub e cole em Chaves no seu perfil do github.

Para acessar o contêiner, use o comando abaixo dentro do terminal da imagem Docker que está executando:

```
git config --global user.name "Seu Nome"
git config --global user.email "seuemail@exemplo.com"
git clone <url-do-repositório>
```

Teste do Node.js:

Crie um simples servidor HTTP com Node.js para testar o ambiente. Dentro da área de execução do contêiner Docker, digite node e, em seguida, .editor para entrar no modo de edição do NodeJs. Cole o código abaixo para criar um servidor de testes :

```
const http = require('http');


http.createServer((req, res) => {
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('Hello World\n');
}).listen(3000, () => {
  console.log('Server running at http://localhost:3000/');
});
```