



Serviço Nacional de Aprendizagem Industrial

PELO FUTURO DO TRABALHO

Criação E Desenvolvimento Com Containers

Professor: Henrique Delegrego

Criação E Desenvolvimento Com Containers

Docker Image

- Artefato executável da aplicação que contém o código fonte junto com todas as dependências, configurações e ferramentas necessárias
- Inclui o código do aplicativo, variáveis de ambiente, bibliotecas e dependências
- Composta por uma série de camadas, cada uma representando um conjunto de alterações no sistema de arquivos (como adicionar arquivos, instalar pacotes ou modificar configurações)
- É possível criar uma imagem a partir de um **Dockerfile**



Criação E Desenvolvimento Com Containers

Docker Container

- Instância executável de uma imagem
- Ambiente isolado onde a imagem é executada, com uma arquitetura composta de:
 - **Namespaces:** Isolam recursos como processos, interfaces de rede, sistemas de arquivos e IDs de usuário. Cada container opera em seu próprio namespace, sem interferir nos outros
 - **Control Groups (cgroups):** Limitam e monitoram o uso de recursos, garantindo que não interfiram uns nos outros ou sobrecarreguem o host
 - **Runtime:** Interage com o SO e gerencia o ciclo de vida dos containers

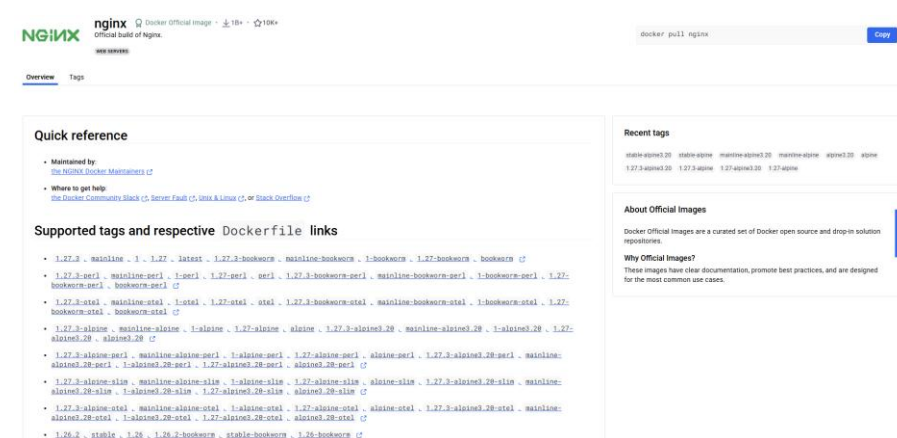
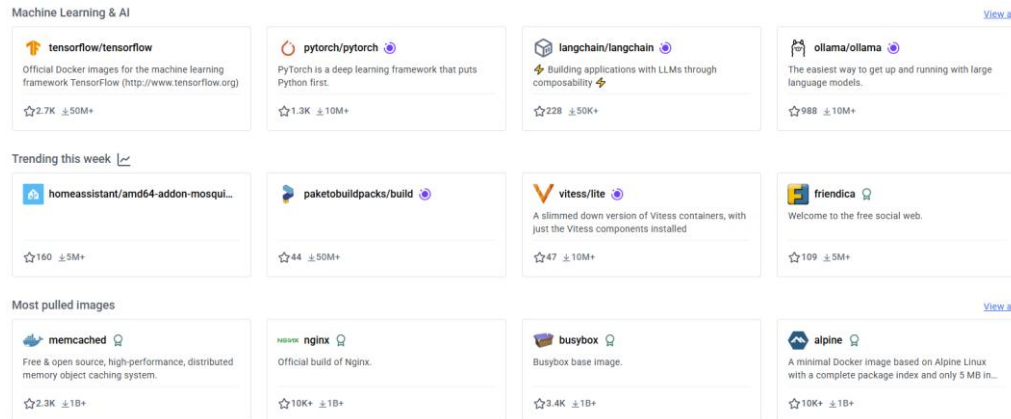
Agora:

- Vamos fazer o nosso Hello World com o Docker

Criação E Desenvolvimento Com Containers

Docker Hub

- Usado para armazenar e distribuir Imagens Docker
- Funciona como uma biblioteca de imagens, permitindo que os usuários façam upload, armazenem, gerenciem e baixem imagens
- As imagens são etiquetadas com identificadores únicos de versão, facilitando o gerenciamento de diferentes versões



Agora:

- Vamos conhecer o Docker Hub

Criação E Desenvolvimento Com Containers

Comandos básicos: pull

- Para fazer o download de uma imagem utilizamos o comando **pull**
- Baixa uma imagem do Docker Hub para o seu armazenamento de imagens local
- Se uma imagem já estiver presente localmente, verifica se há atualizações e baixa apenas as camadas alteradas
- O comando é **docker pull nome-imagem[:tag | @digest]**
- Se você omitir a tag ou digest, o Docker usará a tag **latest** por padrão
- É boa prática pegar alguma versão anterior à latest

```
C:\Users\henri>docker pull nginx:1.26
```

Criação E Desenvolvimento Com Containers

Comandos básicos: images

- Para listar as imagens armazenadas localmente, executamos o comando **docker images**

```
C:\Users\henri>docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
nginx         1.26     64e5ac93d424   6 months ago   192MB
```

Agora:

- Pesquisem por algumas imagens no Docker Hub e façam o download delas
- Listem essas imagens

Criação E Desenvolvimento Com Containers

Comandos básicos: run

- Serve para criar e iniciar um novo container a partir de uma imagem especificada
- Usamos o comando **docker run nome-imagem<:tag | @digest>**
- Se a imagem especificada não existir localmente, é feito o download dela

```
C:\Users\henri>docker run nginx:1.26
```


Criação E Desenvolvimento Com Containers

Comandos básicos: ps

- Comando para listar os containers
- Mostra apenas os container em execução
- Use a flag **-a** para incluir containers que não estão em execução

```
C:\Users\henri>docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
69df9c5a4f25	nginx	"/docker-entrypoint..."	About a minute ago	Up About a minute	80/tcp	charming_knuth

Agora:

- Iniciem o container de alguma distribuição de Linux de sua escolha e listem-no

Criação E Desenvolvimento Com Containers

Entrypoint

- Um container só fica em execução enquanto seu processo principal estiver em execução
- Alguns containers, como servidores web são projetados para executar um serviço de longa duração, esses containers geralmente têm um processo em primeiro plano que continua em execução

```
C:\Users\henri>docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
69df9c5a4f25	nginx	"/docker-entrypoint..."	About a minute ago	Up About a minute	80/tcp	charming_knuth

Agora:

- Iniciem o container de algum servidor web
- Notem que o terminal ficou “dedicado” ao container, então temos que abrir outro

Criação E Desenvolvimento Com Containers

Comandos básicos: tags

- **-d**: Executa o container em modo desacoplado, sem travar o terminal
- **--name**: Dá um nome ao container

```
C:\Users\henri>docker run -d --name meu-container nginx:1.26
ad555b859e55c0f041180e511e185e58855b80d24a97b2703673f251b45ea291

C:\Users\henri>docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS        NAMES
ad555b859e55   nginx:1.26 "/docker-entrypoint...." 2 seconds ago  Up 2 seconds  80/tcp       meu-container
```

Criação E Desenvolvimento Com Containers

Comandos básicos: start e stop

- A partir do **id** do container podemos parar e iniciar ele

Docker stop

```
C:\Users\henri>docker stop b5e36645bd0e  
b5e36645bd0e
```

Docker start

```
C:\Users\henri>docker start b5e36645bd0e  
b5e36645bd0e
```

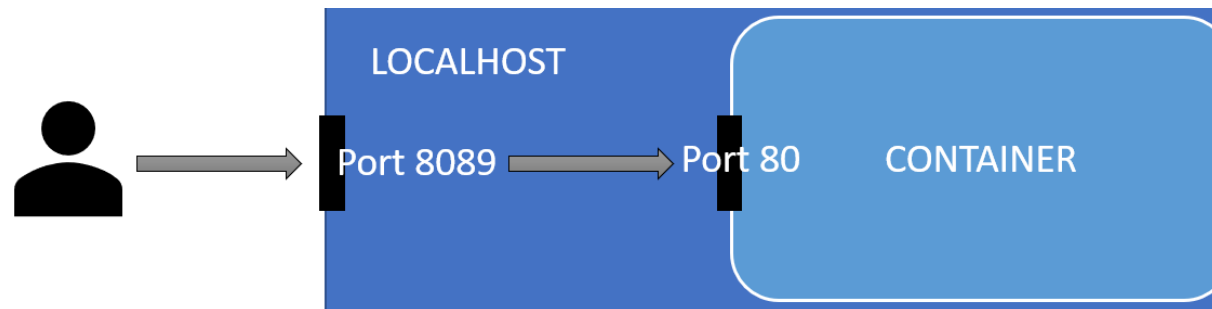
Agora:

- Pratiquem os comandos passados
- Aproveite para conhecer outros comandos
- Ex: **sleep, pause, unpause, rm, rm -force**

Criação E Desenvolvimento Com Containers

Port Binding

- Um container quando criado roda na rede interna do Docker
- Port Binding é o ato de publicar uma dessas portas do container no host, de forma que conexões feitas no host cheguem ao container
- **Porta do container:**
 - Porta dentro do container Docker onde um serviço ou aplicação está ouvindo
- **Porta do host:**
 - Porta na máquina host que redireciona o tráfego para a porta do container



Criação E Desenvolvimento Com Containers

Port Binding

- O comando é composto pela tag **-p (porta-host:porta-container)**

```
C:\Users\henri>docker run -d -p 4000:80 nginx:1.26
e44c2d1fcc7ce657fdcc060ee77a7ac38cf0ac6f5b641f03d36229ee8586bbd7

C:\Users\henri>docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                               NAMES
e44c2d1fcc7c   nginx:1.26 "/docker-entrypoint..." 3 seconds ago  Up 2 seconds  0.0.0.0:4000->80/tcp              stoic_spence
```

- Somente um serviço pode ser executado em uma porta de host específica
- É uma prática comum mapear a porta do host para a mesma porta exposta pelo container

Agora:

- Iniciem algum servidor web em uma porta do host

Criação E Desenvolvimento Com Containers

Variáveis de Ambiente

- Variáveis de ambiente são pares chave-valor que os processos dentro do container podem acessar
- Essas variáveis estão descritas na documentação do repositório da imagem
- Definidas no momento em que você inicia um container utilizando a flag **-e**

```
C:\Users\henri>docker run -e MYSQL_ROOT_PASSWORD=senha -p 3306:3306 -d mysql
```

Agora:

- Fazer a atividade de importância do Docker



Serviço Nacional de Aprendizagem Industrial

PELO FUTURO DO TRABALHO

0800 048 1212     **sc.senai.br**

Rodovia Admar Gonzaga, 2765 - Itacorubi - 88034-001 - Florianópolis, SC