

# DEVOPS

DIA 6 DE 10





**Augusto Nunes**  
PSM, MCT, OCA, OCP, MCSD



**augustonunes**



**Augusto Nunes**



**augusto.cbn**

# DOCKER



# DOCKER – SUMÁRIO

---

- Visão Geral
- Principais Comandos
- Laboratórios





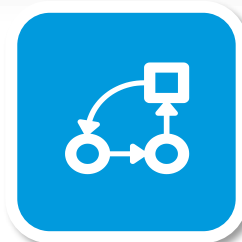
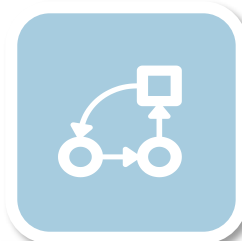
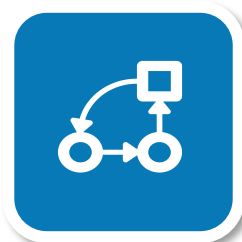
# DOCKER

## VISÃO GERAL

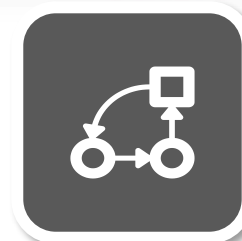
# DOCKER - VISÃO GERAL

Formato de imagem de contêiner.

Ferramenta de linha de comando para empacotar e executar contêineres.



API para gerenciamento de contêineres.

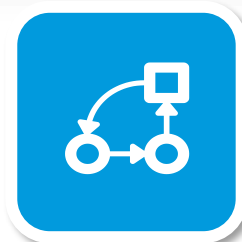
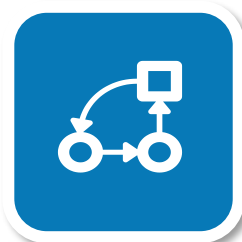


Runtime de contêiner que administra o ciclo de vida de contêiner.

# DOCKER – IMAGEM DE CONTÊINER

Armazena tudo que é necessário para executar um contêiner.

Arquivo compactado.

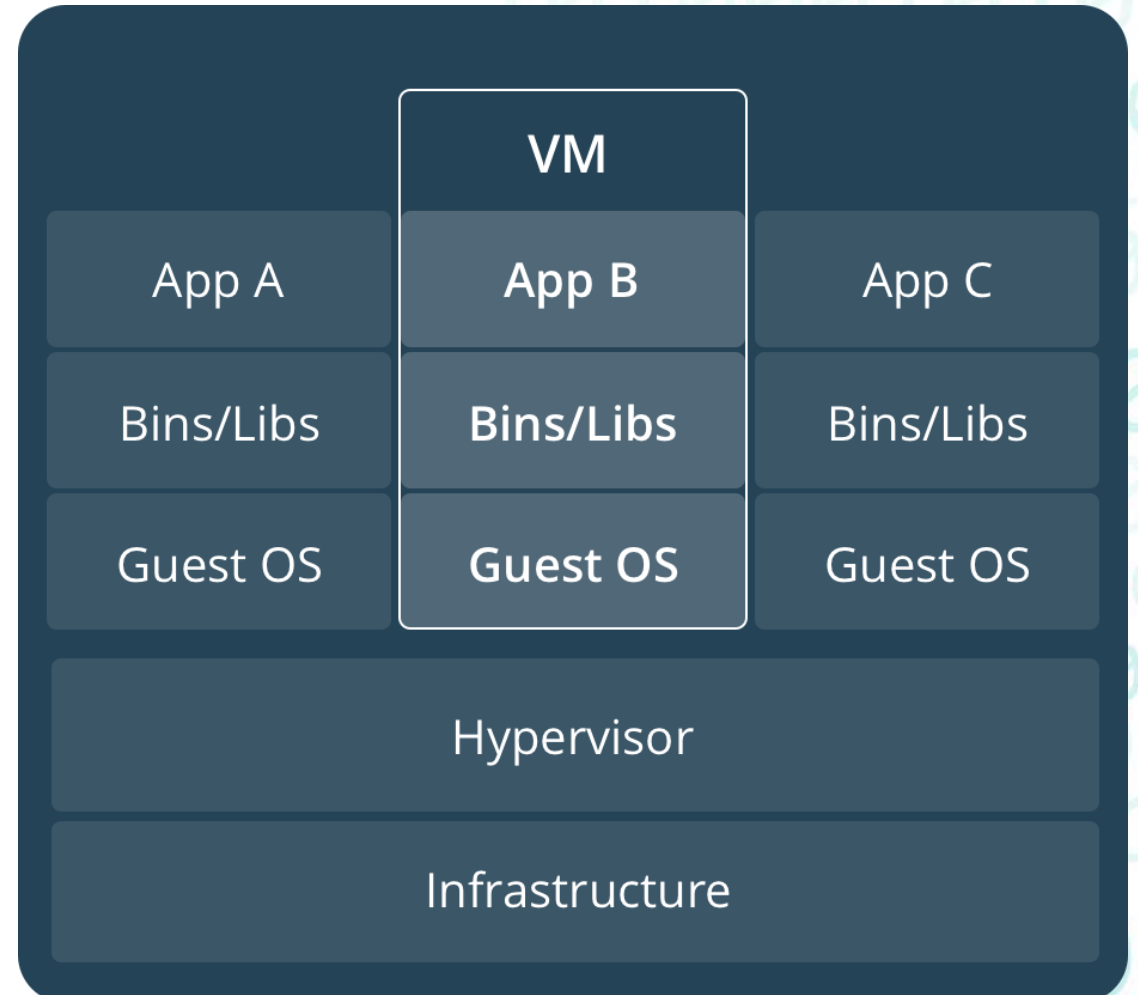
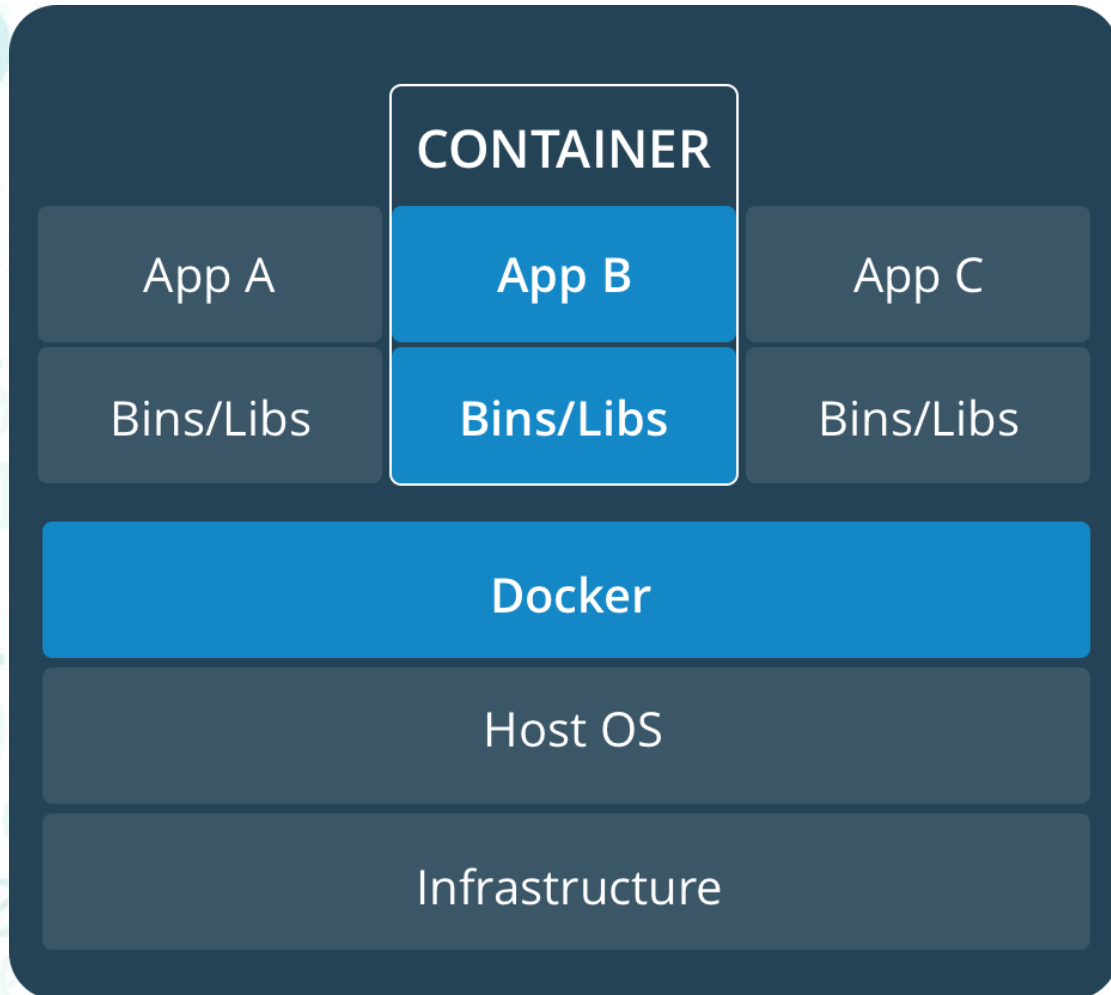


Arquivo binário único, com um identificador exclusivo.

Fácil distribuição.



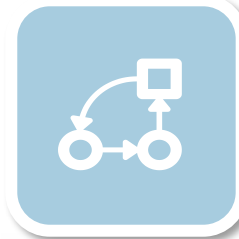
# DOCKER – CONTÊNER VS VIRTUAL MACHINE



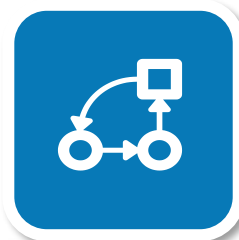


# DOCKER – CONTÊINER VS VIRTUAL MACHINE

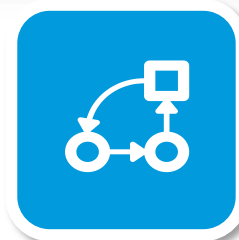
Os contêineres são executados nativamente em Linux. Eles compartilham o Kernel do host com outros contêineres.



As VMs executam um SO completo (Guest), que acessa virtualmente os recursos do Host através do Hypervisor.



A execução de uma VM causa um overhead muito além do que o necessário para executar uma aplicação.



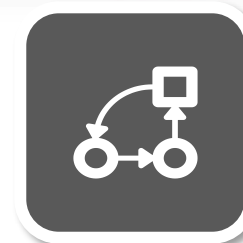
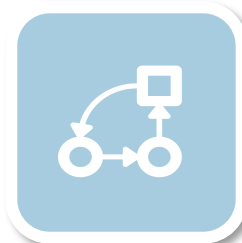
Os contêineres são leves, consomem pouca memória.



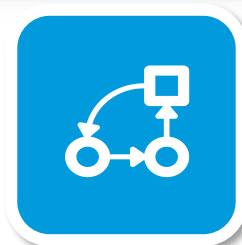
# DOCKER – FILE

Arquivo utilizado na construção de uma imagem de contêiner.

Arquivo de texto.



Arquivo chamado *Dockerfile*.



Arquivo sem extensão.

## DOCKER – FILE

---

```
from mcr.microsoft.com/dotnet/core/runtime:3.1
copy bin/release/netcoreapp3.1/publish/ bin/
workdir /bin
entrypoint ["dotnet", "MegaSena.dll"]
```

**from** imagem base.

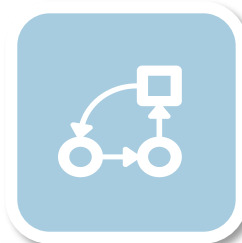
**copy** copia os binários para um diretório específico dentro do container.

**workdir** indica o diretório de trabalho do container

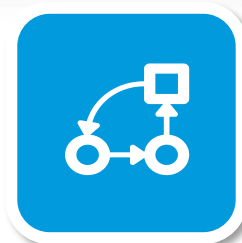
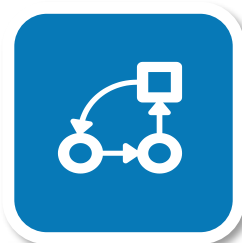
**entrypoint** especifica qual comando será executado quando o container for iniciado.

# DOCKER – HUB

Feed que possibilita o compartilhamento de imagens de contêiner.



Utilizado por projetos open-source e grandes players do mercado de software.



Repositórios públicos e privados.



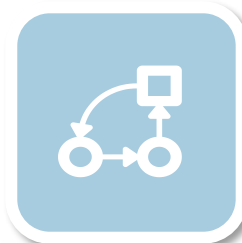
Atualmente o maior repositório do mundo.



# DOCKER – DESKTOP

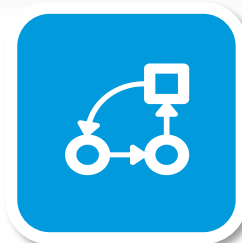
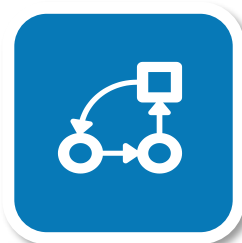
Aplicação com setup simplificado para Mac e Windows.

Ideal para ambientes de desenvolvimento.



Contempla tudo o que é necessário para construir, executar e compartilhar contêineres.

Fácil distribuição.





# DOCKER

## PRINCIPAIS COMANDOS

## DOCKER – PRINCIPAIS COMANDOS – DOCKER VERSION

---

```
c:> docker --version
```

Comando utilizado para checar a instalação do docker e verificar o número do pacote instalado.



## DOCKER – PRINCIPAIS COMANDOS – DOCKER RUN

---

```
c:> docker run  
c:> docker run hello-world
```

Cria e inicia um novo contêiner para a imagem informada.



## DOCKER – PRINCIPAIS COMANDOS – DOCKER IMAGE

---

```
docker image ls
```

Lista as imagens disponíveis na máquina.

## DOCKER – PRINCIPAIS COMANDOS – DOCKER LOGIN

---

```
c:> docker login  
c:> docker login [registry]
```

*[registry]* URL do registro docker

Permite realizar o logon em um registro docker, possibilitando futuras atividades de pull e push.

## DOCKER – PRINCIPAIS COMANDOS – DOCKER PS

---

```
c:> docker ps  
c:> docker ps --all
```

**--all** exibe todos os contêineres, mesmo aqueles que não estão em execução.

Exibe a relação dos contêineres disponíveis.

## DOCKER – PRINCIPAIS COMANDOS – DOCKER BUILD

---

```
c:> docker build -t [nome] -f [nome docker file] .
```

- t nome da imagem
- f nome do arquivo de configuração docker file
- . Path onde o docker file está localizado

Comando responsável por construir uma imagem de contêiner.



## DOCKER – PRINCIPAIS COMANDOS – DOCKER CREATE

---

```
c:> docker create --name [nome do contêiner] [nome da imagem]
```

**--name** permite definir o nome do contêiner

Cria um contêiner a partir de uma image.

## DOCKER – PRINCIPAIS COMANDOS – DOCKER ATTACH

---

```
c:> docker attach [nome do contêiner]
```

**--sig-proxy=false** garante que o contêiner não será encerrado quando o comando **ctrl+c** for executado no terminal.

Permite inspecionar um contêiner em execução.

## DOCKER – PRINCIPAIS COMANDOS – DOCKER START

---

```
c:> docker start [nome do contêiner]
```

Inicia o contêiner parametrizado.

## DOCKER – PRINCIPAIS COMANDOS – DOCKER STOP

---

```
c:> docker stop [nome do contêiner]
```

Encerra o contêiner parametrizado.



## DOCKER – PRINCIPAIS COMANDOS – DOCKER CONTAINER

---

```
c:> docker container rm [nome do contêiner]
```

-f força a remoção do contêiner, mesmo que ele esteja em execução.

Remove o contêiner do repositório local.


## DOCKER – PRINCIPAIS COMANDOS – DOCKER CONTAINER

---

```
c:> docker image rm [id da imagem]
```

-f força a remoção do contêiner, mesmo que ele esteja em execução.

Remove a imagem do repositório local.



# DOCKER

## LABORATÓRIO 6