**Docker**

[**Docker** 1](#_Toc125505294)

[**O que é o Docker** 1](#_Toc125505295)

[**Instalação** 1](#_Toc125505296)

[**Comandos básicos** 1](#_Toc125505297)

[**Conceitos de imagem e contêiner** 2](#_Toc125505298)

[**Repositórios de imagens** 2](#_Toc125505299)

[**Outras funcionalidades** 2](#_Toc125505300)

[**O que é o Docker Desktop** 3](#_Toc125505301)

[**Integração com o Windows** 3](#_Toc125505302)

[**Ferramentas de gerenciamento** 3](#_Toc125505303)

[**Recursos avançados de desenvolvimento** 3](#_Toc125505304)

[**Docker Compose** 3](#_Toc125505305)

[**Docker Swarm** 4](#_Toc125505306)

[**Docker Kitematic** 4](#_Toc125505307)

**O que é o Docker**

Docker é uma plataforma de virtualização de contêineres que permite criar, implantar e executar aplicativos em contêineres.

* Os contêineres são como máquinas virtuais, mas compartilham o kernel do host (sistema operacional) e ocupam menos espaço em disco e memória.
* Isso torna os contêineres mais leves e fáceis de gerenciar do que as máquinas virtuais.

**Instalação**

* Para começar a usar o Docker, você precisará instalar o Docker Engine em seu computador.
* Ele pode ser baixado de forma gratuita no site oficial do Docker.
* Uma vez instalado, você poderá usar o comando "docker" no terminal para gerenciar contêineres.

**Comandos básicos**

* **docker run**: usado para executar um novo contêiner a partir de uma imagem
* **docker ps**: usado para listar os contêiners
* **docker stop**: usado para parar um contêiner em execução
* **docker rm**: usado para remover um contêiner
* **docker pull**: usado para baixar uma imagem de um repositório de imagens
* **docker push**: usado para enviar uma imagem para um repositório de imagens
* **docker build**: usado para construir uma imagem a partir de um arquivo Dockerfile
* **docker images**: usado para listar as imagens baixadas
* **docker exec**: usado para executar comandos dentro de um contêiner em execução
* **docker logs**: usado para visualizar os logs de um contêiner
* **docker inspect**: usado para visualizar informações detalhadas sobre um contêiner ou imagem.

**Conceitos de imagem e contêiner**

* Uma imagem é o modelo de um contêiner e contém tudo o que é necessário para executar uma aplicação (como arquivos, configurações, etc.).
* Um contêiner é uma instância de uma imagem em execução.

**Repositórios de imagens**

Existem diversos repositórios de imagens disponíveis, como o Docker Hub, onde é possível encontrar e baixar imagens prontas para uso.

* Também é possível criar suas próprias imagens a partir de arquivos "Dockerfile" que contenha as instruções para construir a imagem.

**Outras funcionalidades**

* Volumes: os volumes do Docker permitem compartilhar arquivos entre o host e o contêiner, ou entre contêineres.
* Redes: os contêineres Docker podem ser conectados em redes virtuais, permitindo que os contêineres se comuniquem entre si e com o host.
* Compose: o Docker Compose é uma ferramenta que permite definir e executar aplicativos compostos por vários contêineres.
* Swarm: o Docker Swarm é um cluster de gerenciamento de contêineres que permite escalar e gerenciar contêineres em vários hosts.
* Gerenciamento de recursos: o Docker permite limitar recursos como CPU, memória e banda de rede para contêineres, garantindo que eles não consumam mais recursos do que o necessário.
* Segurança: o Docker fornece vários recursos de segurança para proteger os contêineres e os dados armazenados neles, incluindo criptografia de dados e autenticação.
* Integração com outras ferramentas: o Docker pode ser facilmente integrado com outras ferramentas de gerenciamento de contêineres e orquestração, como Kubernetes ou Mesos, para escalar e gerenciar aplicativos em grande escala.
* Imagens oficiais: o Docker Hub possui uma variedade de imagens oficiais que são mantidas pela comunidade e pela equipe do Docker, essas imagens são consideradas estáveis e confiáveis para uso em produção.
* Dockerfiles: são arquivos que contém as instruções para criação de uma imagem, esses arquivos podem ser compartilhados e usados para criar uma imagem personalizada.
* Docker Machine: é uma ferramenta que permite criar e gerenciar máquinas virtuais Docker, é útil para criar ambientes de desenvolvimento e teste em máquinas virtuais.

**O que é o Docker Desktop**

* Docker Desktop é uma plataforma que permite desenvolver, executar e gerenciar contêineres no Windows.
* Inclui o Docker Engine, Docker Compose e Docker Swarm.
* Ferramentas adicionais como o Docker Kitematic e recursos avançados para desenvolvimento.

**Integração com o Windows**

* Docker Desktop é projetado para funcionar diretamente no Windows, sem a necessidade de uma máquina virtual adicional.
* Isso é possível graças ao recurso do Windows chamado Hyper-V.
* Ao instalar o Docker Desktop no Windows, ele habilita o Hyper-V e cria uma máquina virtual chamada "MobyLinuxVM", que é usada para executar os contêineres.
* Isso permite que os contêineres sejam executados em um ambiente isolado, mas compartilhando recursos do host como CPU, memória e rede.

**Ferramentas de gerenciamento**

* O Docker Kitematic é uma interface gráfica de usuário para gerenciar contêineres.
* Ele permite visualizar e gerenciar contêineres, ver logs, visualizar portas expostas e muito mais.
* Ele também inclui a capacidade de gerenciar contêineres e aplicativos usando o Docker Compose.

**Recursos avançados de desenvolvimento**

* O modo de desenvolvimento permite desenvolver aplicativos diretamente no Windows e testá-los em contêineres sem precisar fazer build ou push para um registro.

**Docker Compose**

* Docker Compose é uma ferramenta que permite definir e executar aplicativos compostos por vários contêineres.
* Ele usa arquivos de configuração chamados "docker-compose.yml" para definir os contêineres, volumes, redes e configurações de cada aplicativo.
* Isso permite que você gerencie facilmente vários contêineres como um único aplicativo.
* Com o Compose, é possível executar comandos como "docker-compose up" para iniciar todos os contêineres de um aplicativo, ou "docker-compose down" para pará-los.

**Docker Swarm**

* Docker Swarm é um cluster de gerenciamento de contêineres que permite escalar e gerenciar contêineres em vários hosts.
* Ele permite que você crie um cluster de nós (hosts) e gerencie contêineres em todos os nós de forma centralizada.
* Isso permite escalar aplicativos automaticamente para lidar com o aumento de tráfego, e gerenciar facilmente contêineres em vários hosts.

**Docker Kitematic**

* Docker Kitematic é uma interface gráfica de usuário para gerenciar contêineres.
* Ele permite visualizar e gerenciar contêineres, ver logs, visualizar portas expostas e muito mais.
* Ele também inclui a capacidade de gerenciar contêineres e aplicativos usando o Docker Compose.