

DigitalHouse>

Docker Hub



**Certified Tech
Developer**

The Ultimate Degree



O **Docker Hub** é um **serviço de registro de repositório**

fornecido pela Docker Inc.

Compartilhamento e
colaboração são suas
premissas.



Índice

1. [Para que serve o Docker Hub?](#)
2. [Recursos do Docker Hub](#)
3. [Criação do primeiro repositório](#)
4. [Explore as imagens](#)
5. [Baixar uma imagem](#)
6. [Crie uma imagem](#)
7. [Enviar uma imagem](#)
8. [O que são imagens certificadas pelo Docker?](#)
9. [Imagens populares no Docker Hub](#)

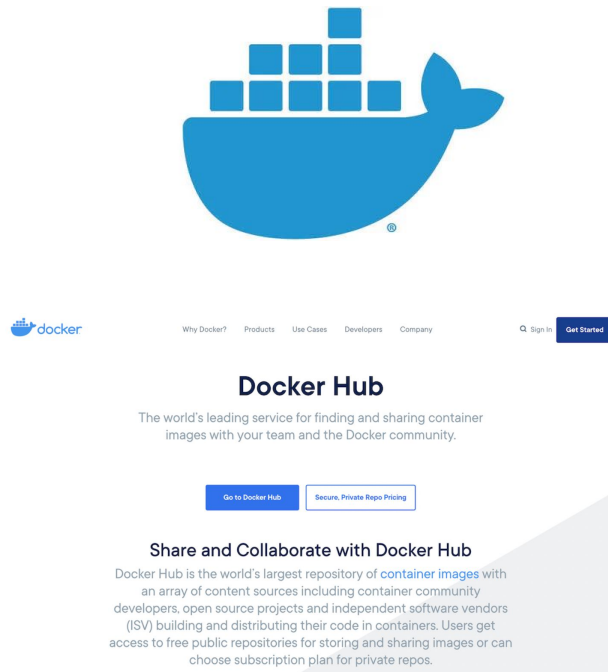
1

**Para que serve o
Docker Hub?**

Para que serve o Docker Hub?

Ele nos permite extrair e enviar imagens do Docker para o Docker Hub. Podemos tratar isso como um GitHub, onde obtemos e enviamos nosso código-fonte, mas no caso do Docker Hub baixamos ou publicamos nossas imagens de containers.

É um repositório online baseado em nuvem que armazena os dois tipos de repositórios, ou seja, repositório público e privado. Os repositórios públicos são acessíveis a todos, mas o privado é acessível ao proprietário interessado dos repositórios. Há também um custo associado se armazenarmos mais de um certo número de repositórios como privados.

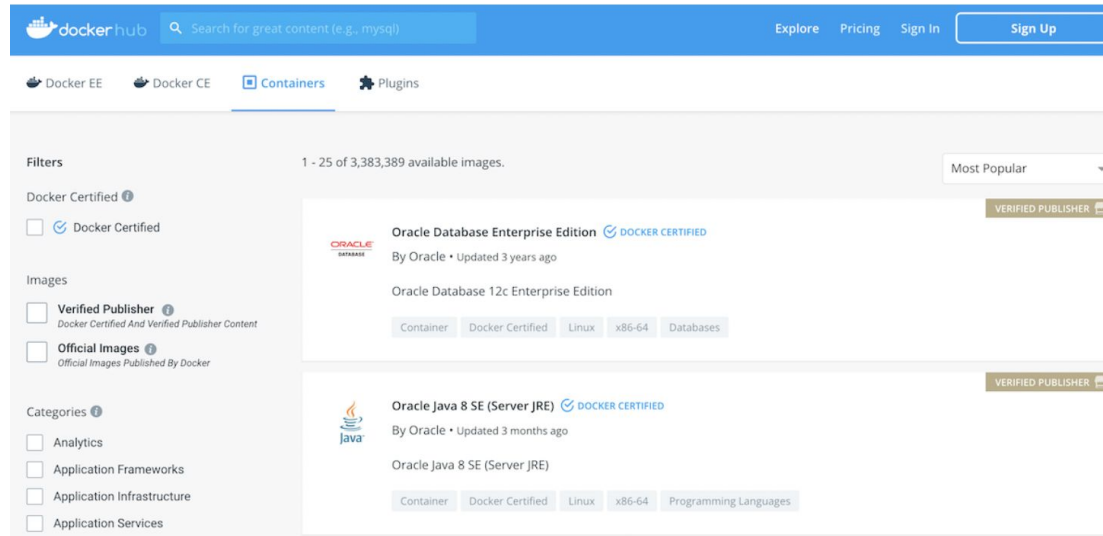


2 | Recursos do Docker Hub

Recursos do Docker Hub

1. Repositórios de imagens:

Ele nos ajuda a encontrar e extrair imagens de container do Docker Hub.



Recursos do Docker Hub



2. Equipe e organizações:

Ele nos permite criar grupos de trabalho e direcionar repositórios privados, que estão disponíveis para uso apenas dentro de nossa organização. Dessa forma, gerenciamos o acesso aos nossos repositórios de imagens de containers privados.



3. Integração GitHub e Bitbucket:

Permite integração com repositórios de código-fonte, como GitHub e Bitbucket.



4. Compilações automatizadas:

Se alguma alteração no código-fonte tiver sido enviada para repositórios, ele detectará e criará automaticamente imagens de container do GitHub ou Bitbucket e as enviará para o Docker Hub.

Recursos do Docker Hub



5. Webhooks:

Depois de enviarmos nossas imagens com sucesso, com a ajuda de um webhook, ele acionará uma ação para integrar o Docker Hub a outros serviços.



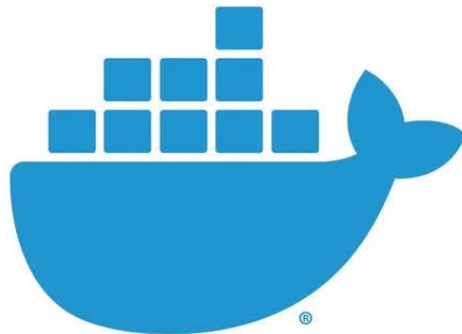
6. Imagens oficiais e da editora:

Imagens de alta qualidade fornecidas por dockers são consideradas imagens oficiais e podem ser extraídas e usadas. Da mesma forma, imagens de alta qualidade fornecidas por fornecedores terceirizados são imagens de editores, também chamadas de imagens certificadas, que fornecem suporte e garantia de compatibilidade com o Docker Enterprise.

3 | Criação do primeiro repositório

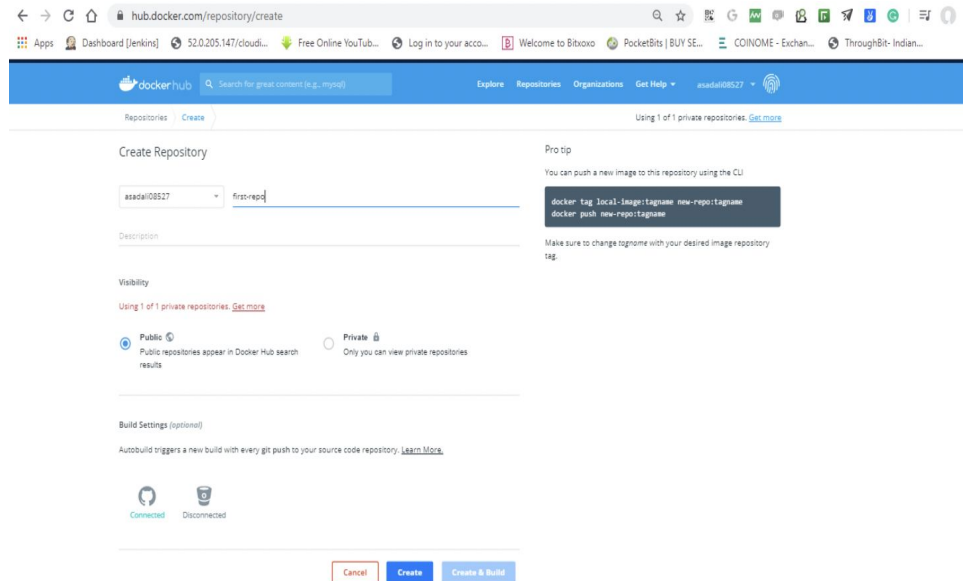
Criação do primeiro repositório

Esta etapa requer o login no Docker Hub usando suas credenciais de login. Caso não tenha uma conta, você pode criar uma clicando no botão **“Cadastre-se”**, disponível no site.



Uma vez logado, você pode criar o repositório clicando em “**Criar repositório**” na página de boas-vindas.

Em seguida, ele solicitará um nome e dará um ao seu repositório.



The screenshot shows the Docker Hub 'Create Repository' page. The browser address bar shows 'hub.docker.com/repository/create'. The page has a blue header with the Docker Hub logo and navigation links: Explore, Repositories, Organizations, Get Help, and a user profile 'asad0108527'. Below the header, there's a section titled 'Create Repository'. It includes a dropdown menu for the namespace (currently 'asad0108527') and a text input for the repository name (currently 'first-repo'). A 'Description' field is also present. Under the 'Visibility' section, the 'Public' radio button is selected, with a note that public repositories appear in Docker Hub search results. The 'Private' option is also available. Below this, there's a 'Build Settings (optional)' section with a note about autobuild and a 'Learn More' link. At the bottom, there are status indicators for 'Connected' and 'Disconnected' with corresponding icons. Finally, at the very bottom, there are three buttons: 'Cancel', 'Create', and 'Create & Build'.

hub.docker.com/repository/create

Apps Dashboard [Jenkins] 52.0.205.147/cloud... Free Online YouTube... Log in to your acco... Welcome to Bitboxo PocketBets | BUY SE... COINOME - Exchan... ThroughBit- Indian...

docker/hub Search for great content (e.g., mypet) Explore Repositories Organizations Get Help + asad0108527

Repositories Create

Using 1 of 1 private repositories. [Get more](#)

Create Repository

asad0108527 first-repo

Description

Pro tip


You can push a new image to this repository using the CLI


```
docker tag local-image:tagname new-repo:tagname
docker push new-repo:tagname
```

Make sure to change tagname with your desired image repository tag.

Visibility



Using 1 of 1 private repositories. [Get more](#)

☒ Public  Public repositories appear in Docker Hub search results

☐ Private  Only you can view private repositories

Build Settings (optional)

Autobuild triggers a new build with every git push to your source code repository. [Learn More](#)

 Connected  Disconnected

Cancel Create Create & Build

Você pode selecionar uma **visibilidade pública** ou **privada**. Você também pode integrar seus repositórios de código-fonte, como **GitHub** e **Bitbucket**, por meio da configuração de compilação, mas isso é opcional (e também pode ser feito em um estágio posterior).

Feito tudo, clique em **“Criar”**. Com essas etapas, você pode criar seus primeiros repositórios. No próximo slide você pode ver como eles ficarão.



O Docker Hub nos oferece apenas **um repositório privado gratuito**. Se precisarmos de mais repositórios privados, podemos atualizar nossa conta para um plano pago.

Com a ferramenta de terminal do **Docker Desktop** baixada e instalada, podemos fazer login no Docker Hub usando um comando.

A dark-themed terminal window with a vertical bar on the left. The text 'docker login' is displayed in a pinkish-red monospace font.

```
docker login
```

4 | Explorar as imagens

Explore as imagens

Há duas maneiras de pesquisar imagens e repositórios públicos no Docker Hub. Você pode pesquisá-lo no **site do Docker Hub** ou usar a **ferramenta de linha de comando** e executar o seguinte comando, supondo que queremos pesquisar a imagem do repositório MySQL.

```
docker search mysql
```

```
root@ip-xxx-xx-xx-xx:~# docker search mysql
```

NAME	DESCRIPTION	STARS	OFFICIAL	AUTOMATED
mysql	MySQL is a widely used, open-source relation...	9417	[OK]	
mariadb	MariaDB is a community-developed fork of MyS...	3392	[OK]	
mysql/mysql-server	Optimized MySQL Server Docker images. Create...	688		[OK]
centos/mysql-57-centos7	MySQL 5.7 SQL database server	74		
mysql/mysql-cluster	Experimental MySQL Cluster Docker images. Cr...	66		
centurylink/mysql	Image containing mysql. Optimized to be link...	61		[OK]
deitch/mysql-backup	REPLACED! Please use http://hub.docker.com/r...	41		[OK]
bitnami/mysql	Bitnami MySQL Docker Image	39		[OK]
tutum/mysql	Base docker image to run a MySQL database se...	34		
schickling/mysql-backup-s3	Backup MySQL to S3 (supports periodic backup...	29		[OK]
prom/mysqld-exporter		27		[OK]
linuxserver/mysql	A Mysql container, brought to you by LinuxSe...	25		
centos/mysql-56-centos7	MySQL 5.6 SQL database server	19		
circleci/mysql	MySQL is a widely used, open-source relation...	19		
databack/mysql-backup	Back up mysql databases to... anywhere!	15		
mysql/mysql-router	MySQL Router provides transparent routing be...	15		
arey/mysql-client	Run a MySQL client from a docker container	13		[OK]
openshift/mysql-55-centos7	DEPRECATED: A Centos7 based MySQL v5.5 image...	6		
fradelg/mysql-cron-backup	MySQL/MariaDB database backup using cron tas...	6		[OK]
genschsa/mysql-employees	MySQL Employee Sample Database	5		[OK]
devilbox/mysql	Retagged MySQL, MariaDB and PerconaDB offici...	3		
ansibleplaybookbundle/mysql-apb	An APB which deploys RHSCl MySQL	2		[OK]
jelastic/mysql	An image of the MySQL database server mainta...	1		
widdpim/mysql-client	Dockerized MySQL Client (5.7) including Curl...	0		[OK]
monasca/mysql-init	A minimal decoupled init container for mysql	0		

5 | Baixar uma imagem

Baixar uma imagem

Podemos baixar uma imagem do Docker Hub usando o comando **pull** da seguinte forma:

```
# docker pull mysql
```

Se já temos **mysql image** em nossa máquina, o comando acima atualizará automaticamente a imagem para a versão mais recente. Uma coisa a notar aqui é que, se notarmos a saída do comando `docker search`, há muitas imagens **MySQL** no **Docker Hub**, e isso porque qualquer pessoa pode enviar uma imagem. Mas cabe a nós saber qual usar com base em nosso caso de uso e garantir que seja apropriado.

Digamos que queremos extrair uma imagem **bitnami/mysql**:

```
# docker pull bitnami/mysql
```

6 | Crie uma imagem

Crie uma imagem

O processo de criação de uma imagem requer um Dockerfile. Podemos pensar em um Dockerfile como um **manual de instruções** que informa ao Docker o que montar. Resumindo, é um arquivo de configuração que segue as instruções de montagem.

Como funciona?

O Docker lê instruções de um Dockerfile e cria imagens automaticamente. A imagem do Docker é um sistema de arquivos em camadas e consiste em várias camadas de somente leitura. Cada camada de uma imagem do Docker representa as instruções de um Dockerfile. Em seguida, vamos percorrer as etapas para criar uma imagem usando um Dockerfile que especifica a configuração do nosso aplicativo.

```
# sudo vim Dockerfile
```

Nota: o nome do arquivo deve ser Dockerfile com “D” maiúsculo.

```
FROM ubuntu:16.04
ENV DEBIAN_FRONTEND noninteractive
MAINTAINER someuser@somedomain.com
RUN apt-get update
RUN apt-get install mysql-server -y
CMD echo "My first image created."
```

Vamos dar uma olhada em algumas das palavras-chave importantes usadas no Dockerfile. Podemos usar # para adicionar um comentário a um Dockerfile.

FROM (de): define a imagem base a ser utilizada.

MAINTAINER (mantenedores): pessoa que vai manter essa imagem.

RUN - Usado para executar a instrução dada para a imagem. No nosso caso, ele primeiro atualiza o sistema e depois instala o MySQL.

CMD - Usado para executar um comando após o lançamento do container.

COPY - Usado para copiar um arquivo do nosso sistema operacional host para o container docker.

EXPOSE - Usado para especificar o número da porta na qual o container executará seu processo.

Depois que nosso Dockerfile for criado com sucesso, precisamos executar o **docker build** para “compilar” nossa imagem localmente e, em seguida, enviá-la para o Docker Hub. Devemos executar este comando dentro da pasta onde o Dockerfile está localizado.

```
docker build ./ -t asadali08527/first-repo
```

Podemos verificar se a imagem foi criada com a seguinte linha de código:

```
docker image ls
```

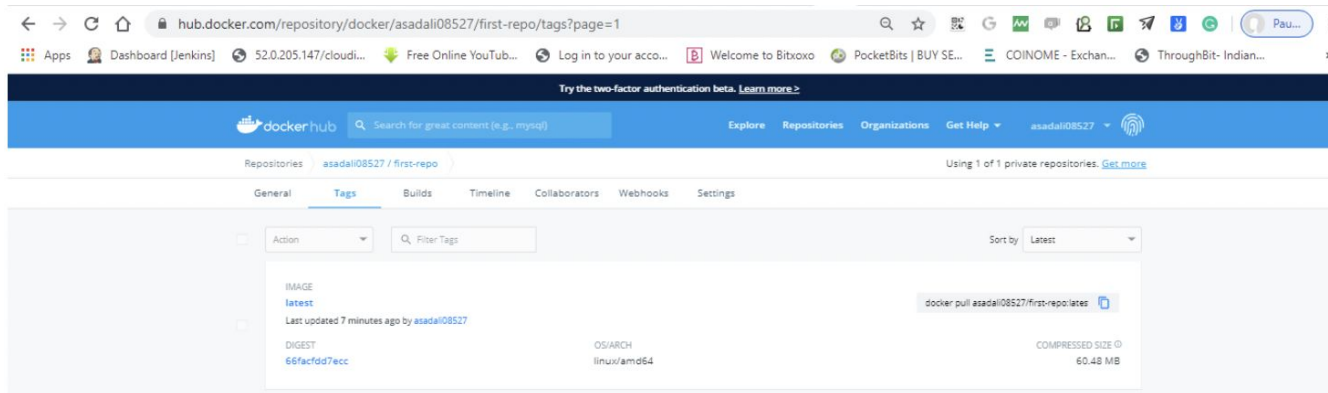
7 | Enviar uma imagem

Enviar uma imagem

Depois que nossa imagem for criada e executada com êxito, podemos enviá-la para o Docker Hub usando o comando push.

```
docker push asadali08527/first-repo
```

Podemos verificar as tags e o status da imagem no Docker Hub, que ficará assim.

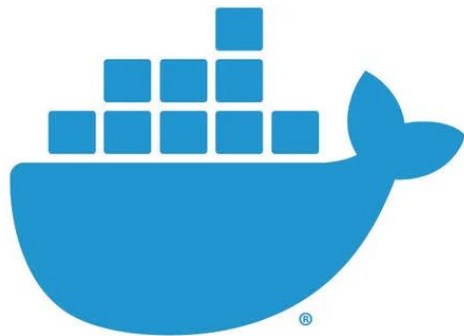


8 | O que são imagens certificadas por Docker?

O que são imagens certificadas pelo Docker?

Estas são as **imagens oficiais** enviadas por fornecedores ou contribuidores. Uma imagem só pode ser certificada pelo Docker Hub se seu conteúdo estiver em conformidade com as **regras, padrões e leis fornecidas pelo Docker Hub**. Em suma, essa imagem deve passar por certos testes de benchmark.

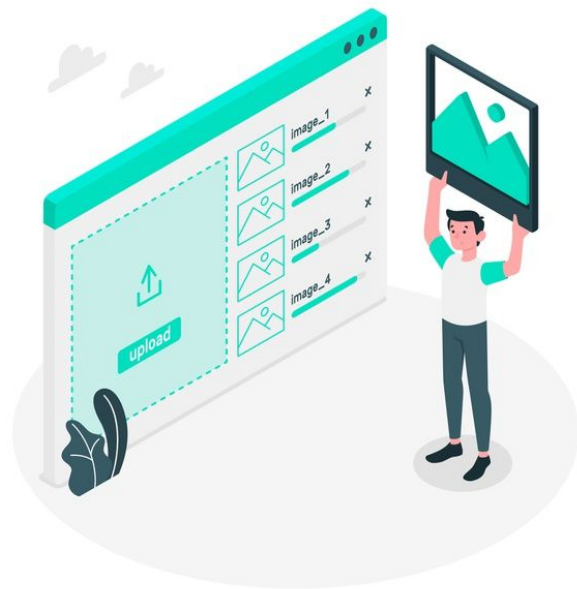
O Docker Hub fornece **inspectDockerImage**, uma ferramenta por meio da qual imagens e plugins podem ser autocertificados por um fornecedor (geralmente o fornecedor ou colaborador publica seus plugins para registrar volumes e redes).



9 | **Imagens populares no Docker Hub**

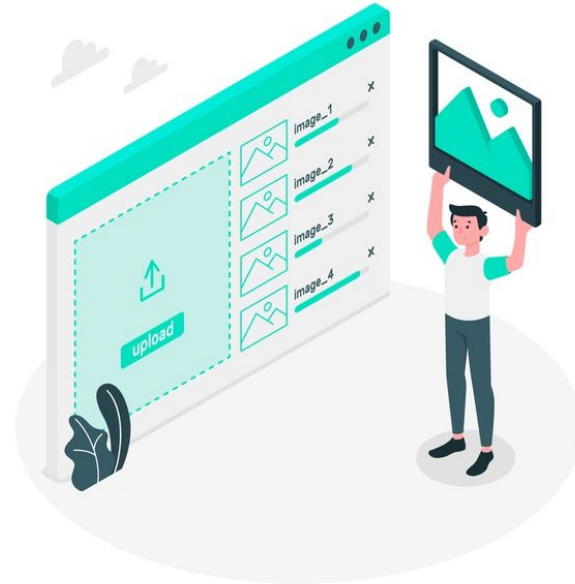
Imagens populares no Docker Hub

Existem muitas imagens selecionadas e otimizadas disponíveis no Docker Hub. A popularidade dessas imagens depende de vários fatores: **presença no mercado, classificações, pontuações de satisfação, entre outros.** Para obter uma lista detalhada dos repositórios mais populares, você pode visitar o site do Docker Hub. O uso de uma imagem também depende do sistema operacional e de sua arquitetura.



Se soubermos para qual sistema operacional e arquitetura as imagens serão usadas, devemos considerar os seguintes fatores-chave antes de extrair uma imagem:

- Procure uma versão específica usando tags (principalmente a mais recente).
- Escolha aquela com o número máximo de downloads e estrelas.
- Encontre a data de sua última atualização.
- Se possível, verifique seu tipo, seja do editor verificado ou oficial (certificado pelo Docker).



DigitalHouse>