

## Preparando o ambiente: Instalando Docker no Ubuntu

Neste passo-a-passo, será visto como instalar o Docker no Ubuntu 64 bits. Todos os comandos listados devem ser executados em seu terminal.

Antes de mais nada, remova possíveis versões antigas do Docker:

```
sudo apt-get remove docker docker-engine docker.io
```

Depois, atualize o banco de dados de pacotes:

```
sudo apt-get update
```

Agora, adicione ao sistema a chave GPG oficial do repositório do Docker:

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

Adicione o repositório do Docker às fontes do APT:

```
sudo add-apt-repository \  
"deb [arch=amd64] https://download.docker.com/linux/ubuntu \  
$(lsb_release -cs) \  
stable"
```

Atualize o banco de dados de pacotes, para ter acesso aos pacotes do Docker a partir do novo repositório adicionado:

```
sudo apt-get update
```

Por fim, instale o pacote docker-ce:

```
sudo apt-get install docker-ce
```

Caso você queira, você pode verificar se o Docker foi instalado corretamente verificando a sua versão:

```
sudo docker version
```

E para executar o Docker sem precisar de `sudo`, adicione o seu usuário ao grupo `docker`:

```
sudo usermod -aG docker $(whoami)
```

## Realizando os testes iniciais

Com o Docker instalado no nosso sistema operacional, já podemos testá-lo para ver o seu funcionamento.

Conseguimos executar os comandos através do terminal nativo do Linux então, vamos abrir um terminal que consiga se comunicar com o nosso Docker, e executar o seguinte comando para verificar a sua versão:

```
docker version
```

Também podemos executar o clássico Hello World:

```
docker run hello-world
```

Ao executar o comando, a primeira mensagem impressa é:

```
Unable to find image 'hello-world:latest' locally -> Não encontrou a imagem!!!  
latest: Pulling from library/hello-world -> Baixou a imagem do Docker Hub
```

Ou seja, o Docker não conseguiu achar a imagem localmente, e ele foi em algum lugar e a baixou. Como assim? Quando executamos o comando **docker run hello-world**, estamos dizendo para o Docker criar um container com a imagem do hello-world. Como não possuímos essa imagem localmente, ele foi buscá-la no Docker Hub, repositório do próprio Docker com várias imagens para utilizarmos em nossos projetos.

Baixada a imagem, ela é executada, exibindo a seguinte mensagem:

```
Hello from Docker!
```

```
This message shows that your installation appears to be working correctly.
```

```
To generate this message, Docker took the following steps:
```

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

```
To try something more ambitious, you can run an Ubuntu container with:
```

```
$ docker run -it ubuntu bash
```

```
Share images, automate workflows, and more with a free Docker ID:
```

```
https://cloud.docker.com/
```

```
For more examples and ideas, visit:
```

```
https://docs.docker.com/engine/userguide/COPIAR CÓDIGO
```

Na mensagem, é detalhado o que foi feito para a execução da imagem. O nosso Docker local entrou em contato com a Docker Engine, que por sua vez baixou a imagem hello-world do Docker Hub, que criou um container com ela e a executou. Após isso, a saída é impressa para nós e a imagem é encerrada.