

Instruções dos programas para a parte prática de Docking

Criado por Bruno Silva e Lucianna Santos – 23/11/21

Para realizar o procedimento de *docking* na aula prática precisamos essencialmente de três programas:

- PyMOL
- MGLTools
- Autodock Vina

A preferência de sistema operacional para instalação desses programas, com exceção do PyMOL, é o Linux. Porém, todos eles possuem instaladores ou instruções de como instalar em ambos os sistemas operacionais, Windows e MacOS.

Porém, para facilitar a prática e deixar todo mundo com o mesmo estilo de instalação, iremos usar o Docker. Com o Docker, é possível lidar com os containers como se fossem máquinas virtuais modulares e extremamente leves. As ferramentas de container, incluindo o Docker, fornecem um modelo de implantação com base em imagem. Isso facilita o compartilhamento de uma aplicação ou conjunto de serviços, incluindo todas as dependências deles em vários ambientes.

Portanto, para a prática nós vamos utilizar uma imagem já pronta com todos os programas instalados e próprios para a aula (<https://hub.docker.com/r/brunomsilva/dockdocker>). A imagem foi criada em Ubuntu 20 e possui não só os programas necessários, mas outros programas de *docking* como o Autodock4 e Dock6.

Instalação e preparação do Docker em Sistemas MacOS e Windows:

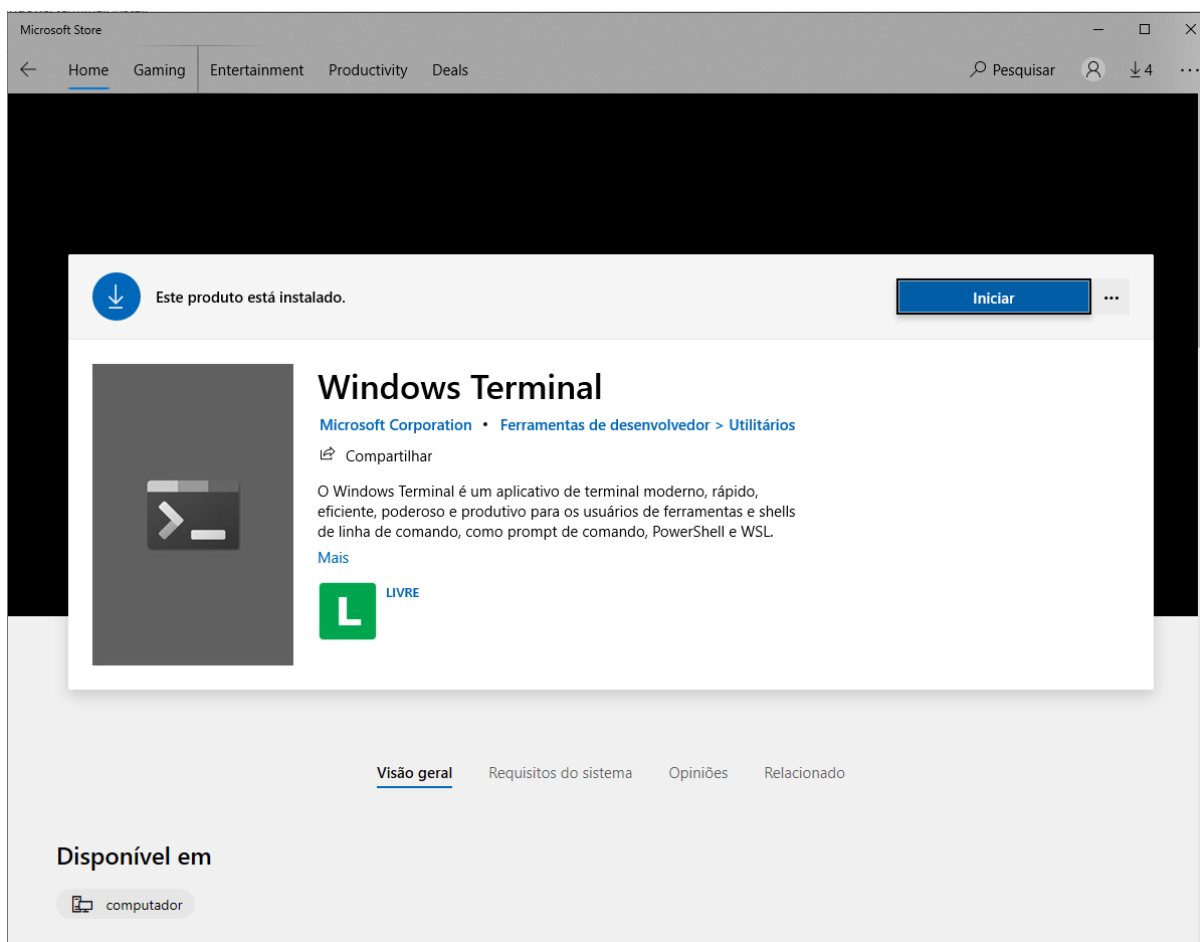
Apesar de ser uma opção fácil de utilizar, alguns programas e passos precisam ser feitos antes de instalar a imagem e utilizar o Docker. Seguimos os seguintes passos:

- 1) Instalar o programa **Docker Desktop**:

<https://www.docker.com/products/docker-desktop>

- 2) Instalar o **Windows Terminal**:

- Abra a Microsoft Store e procure por Windows Terminal;
- Faça o Download e a Instalação pela própria loja.



OBS.: O Windows terminal é uma preferência, pois ele junta o PowerShell, CMD e até mesmo o sistema Ubuntu. Os comandos seguintes podem ser feitos apenas no PowerShell aparentemente.

3) Ativar o **WSL 2** no Windows:

- Abrir o Windows terminal e no PowerShell ou o CMD no modo administrador e digitar:

wsl --install

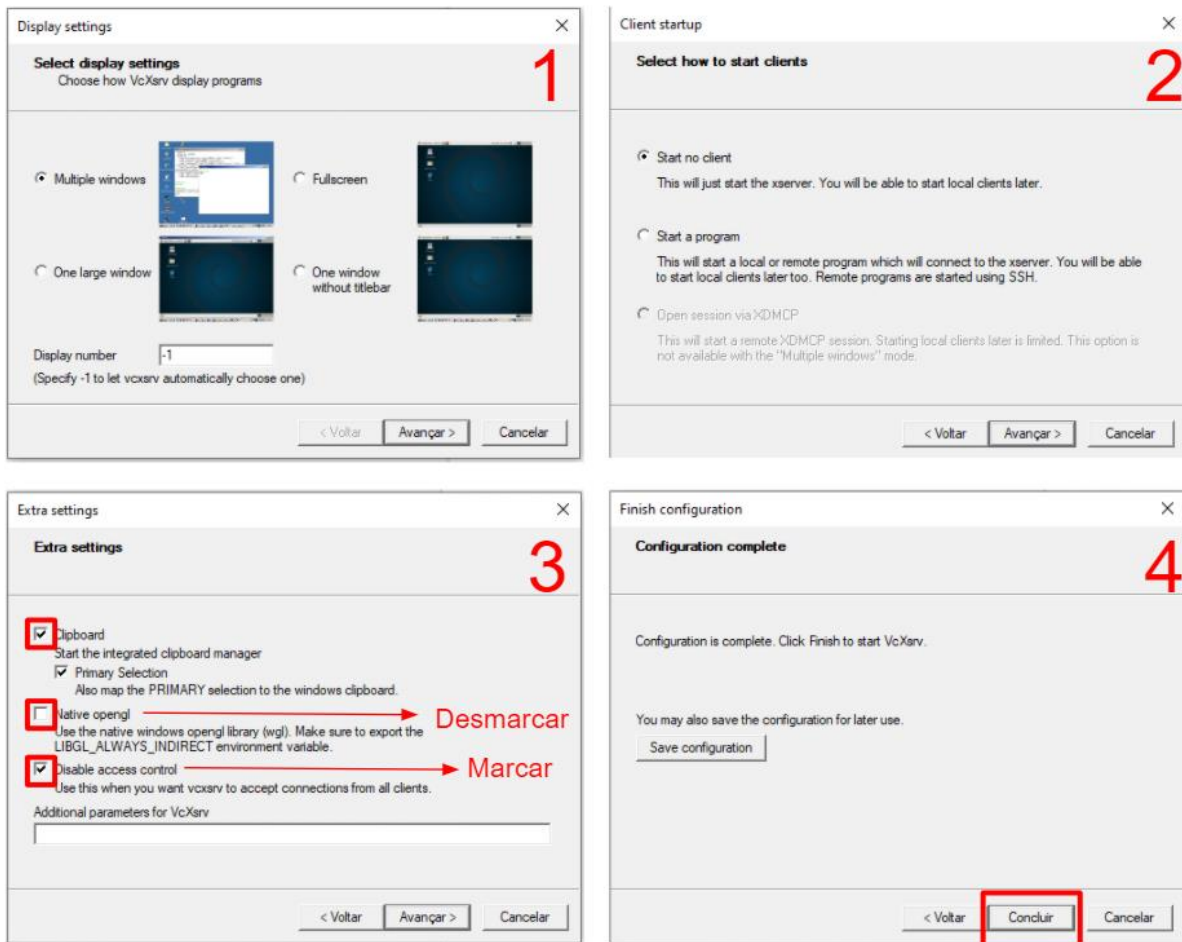
OBS.: Esse procedimento é simples no Windows 10 versão 2004 e superior. Pode ser diferente para outras versões. Ver <https://docs.microsoft.com/pt-br/windows/wsl/install>

4) Instalar o programa **VcXsrv Windows X Server**:

- <https://sourceforge.net/projects/vcxsrv/>

Esse programa auxilia na visualização dos programas gráficos dentro do Docker. No caso, podemos abrir o PyMOL por exemplo.

Depois de instalado, seguir os seguintes passos **TODA** vez que for rodar o Docker para apresentar a parte gráfica:



- 5) Após a instalação dos programas podemos puxar a imagem pronta do Docker que queremos (<https://hub.docker.com/r/brunomsilva/dockdocker>). No **Windows Terminal** digitar:

```
docker pull brunomsilva/dockdocker:latest
```

A seguinte tela vai aparecer, onde a imagem é baixada para o programa Docker.

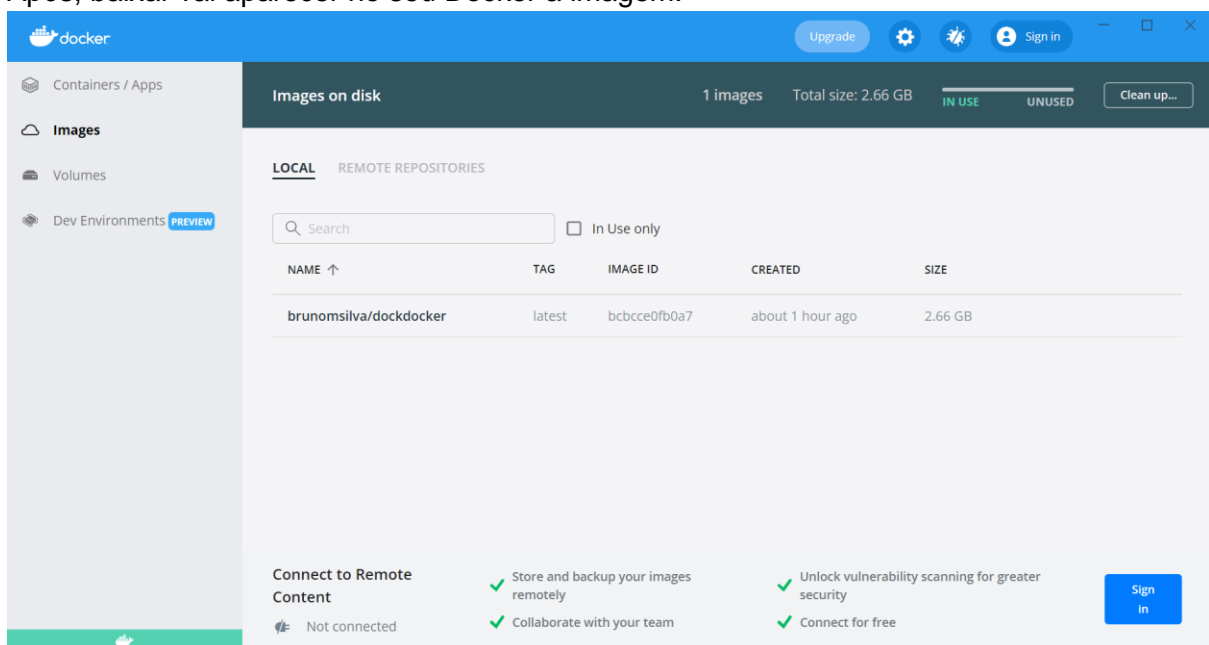
```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Experimente a nova plataforma cruzada PowerShell https://aka.ms/pscore6

PS C:\Users\lucia> docker pull brunomsilva/dockdocker:latest
latest: Pulling from brunomsilva/dockdocker
345e3491a907: Pull complete
57671312ef6f: Pull complete
5e9250ddb7d0: Pull complete
b5ed1ef22699: Downloading [=====>] 153.6MB/293.1MB
7569d2f65e04: Download complete
4f4fb700ef54: Download complete
4ee3589ba7ab: Downloading [=====>] 132.3MB/329.1MB
1bd2d512e83c: Downloading [=====>] 142.8MB/166.1MB
a14deafccc48: Waiting
5250ad85ef6: Waiting
940e816a0018: Waiting
a47e86e78c16: Waiting
bb82b1c2cd98: Waiting
b7e616fb5ec9: Waiting
3a1496956b3c: Waiting
6cc25a16fb24: Waiting
1933dfcc6f6d: Waiting
da643497aff5: Waiting
d8a05919ad75: Waiting
4f06dd7d00f7: Waiting
edc094200e21: Waiting
20d74068c366: Waiting
503cd0270813: Waiting
```

Vai demorar alguns minutos (~10 minutos) para baixar.

Após, baixar vai aparecer no seu Docker a imagem.



A imagem precisa ser baixa apenas essa vez. Depois, precisamos apenas rodar-lá.

- 6) Para rodar o **Docker no Windows**, vamos utilizar o Windows Terminal mais uma vez e digitar:

```
docker run --rm -it -v <caminho/pasta/windows>:/sharreddata -w /sharreddata
brunomsilva/dockdocker:latest bash
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Experimente a nova plataforma cruzada PowerShell https://aka.ms/pscore6

PS C:\Users\lucia> docker run --rm -it -v C:\Users\lucia\Documents\Curso_BSB\pratica:/sharreddata -w /sharreddata brunom
silva/dockdocker:latest bash
```

O <caminho/pasta/windows> no meu caso é uma pasta dentro do Windows chamada C:\Users\lucia\Documents\Curso_BSB\pratica. Nomeando essa pasta eu vou poder ver e armazenar todos os arquivos criados dentro dela. Seria, portanto, uma pasta compartilhada entre o sistema Docker e o Windows.

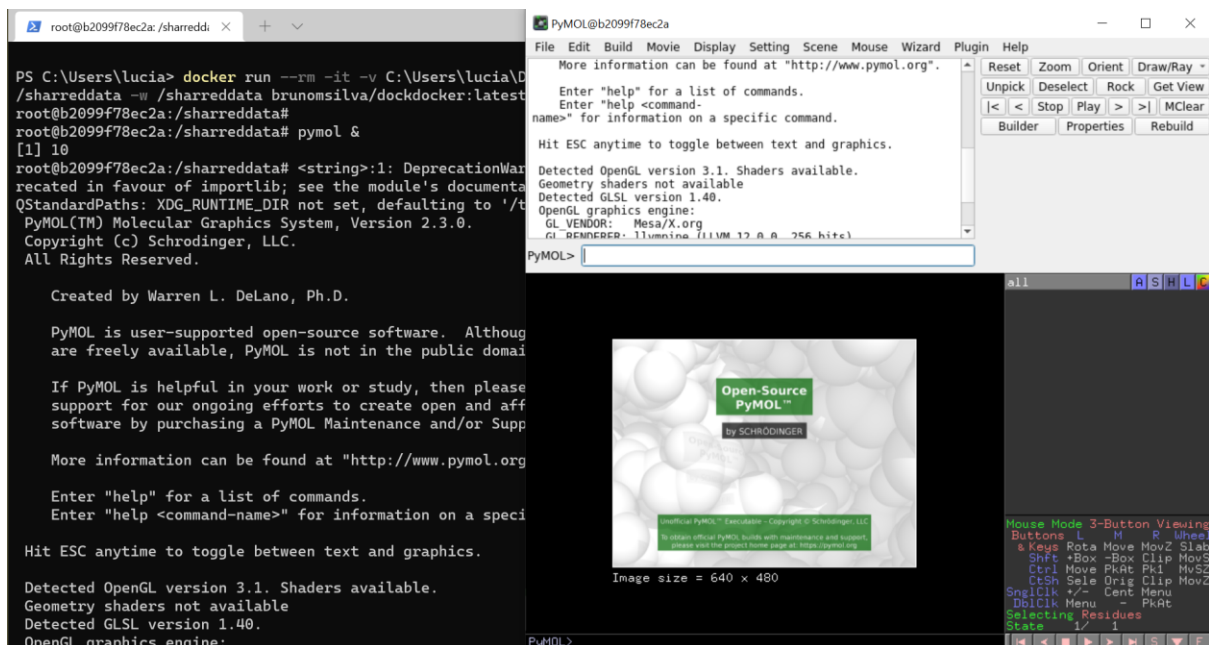
Pronto! Estamos dentro do container Docker.

```
root@b2099f78ec2a:/sharreddata#
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Experimente a nova plataforma cruzada PowerShell https://aka.ms/pscore6

PS C:\Users\lucia> docker run --rm -it -v C:\Users\lucia\Documents\Curso_BSB\pratica:/sharreddata -w /sharreddata brunom
silva/dockdocker:latest bash
root@b2099f78ec2a:/sharreddata#
```

Podemos por exemplo abrir o PyMOL (lembre-se de abrir o Xlauch - VcXsrv Windows X Server antes):



7) Para rodar o Docker em ambientes **MacOS**, digitar:

```
xhost +local:docker
```

```
docker run --rm -it -e DISPLAY=$DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix --
device=/dev/dri -v <caminho/pasta/os>:/sharreddata -w /sharreddata
brunomsilva/dockdocker:latest bash
```

Lembrando que o <caminho/pasta/os> é uma pasta dentro do ambiente que foi criada para a aula por exemplo. Nomeando essa pasta podemos ver e armazenar todos os arquivos criados dentro dela. Seria, portanto, uma pasta compartilhada entre o sistema Docker e o ambiente usado.

8) Para sair do container Docker basta digitar

```
exit
```

No **Terminal**.

Instalação e preparação do Docker em Sistemas Linux (Ubuntu):

<https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-on-ubuntu-18-04>

No **Terminal** do Ubuntu:

- `sudo apt update`
- `sudo apt install apt-transport-https ca-certificates curl software-properties-common`
- `curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -`

- `sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic stable"`
- `sudo apt update`
- `apt-cache policy docker-ce`
- `sudo apt install docker-ce`
- `sudo systemctl status Docker`

Utilizar o Docker sem SUDO:

- `sudo groupadd docker`
- `sudo usermod -aG docker $USER`
- Re-logar no seu usuário
- Digite: **docker run hello-world** para testar se está rodando sem o sudo

Baixar a imagem no ambiente Linux:

Após a instalação dos programas podemos puxar a imagem pronta do Docker que queremos (<https://hub.docker.com/r/brunomsilva/dockdocker>). No **Terminal** digitar:

```
docker pull brunomsilva/dockdocker:latest
```

Rodar o Docker em ambientes Linux:

- `xhost +local:docker`
- `docker run --rm -it -e DISPLAY=$DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix --device=/dev/dri -v <caminho/pasta/windows>:/shareddata -w /shareddata brunomsilva/dockdocker:latest bash`

O <caminho/pasta/os> é uma pasta dentro do ambiente que foi criada para a aula por exemplo. Nomeando essa pasta podemos ver e armazenar todos os arquivos criados dentro dela. Seria, portanto, uma pasta compartilhada entre o sistema Docker e o ambiente usado.