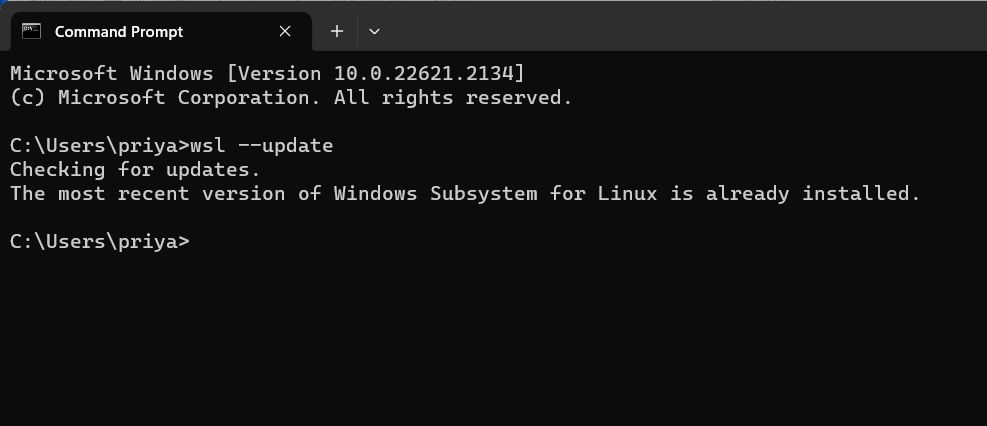
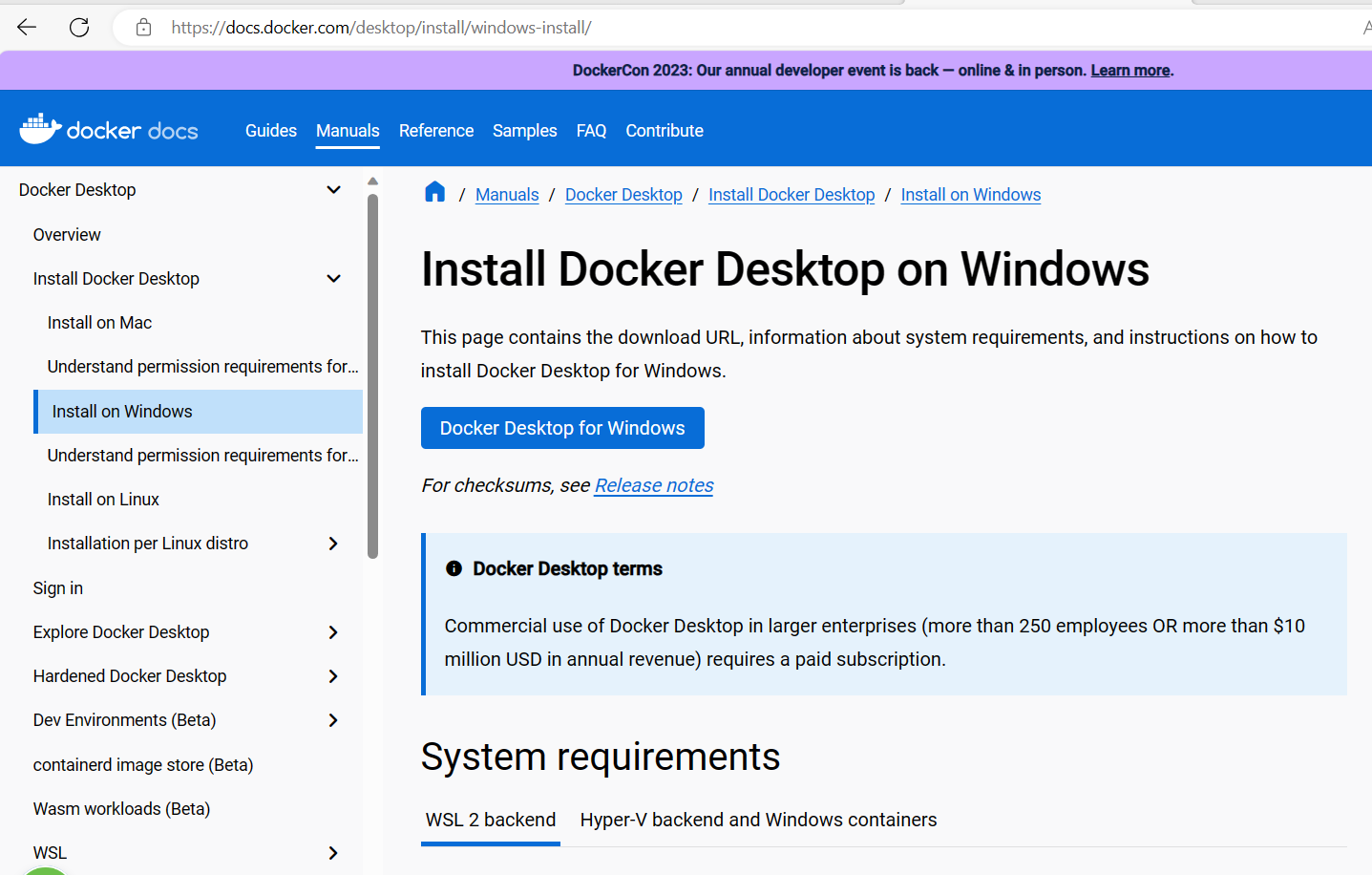
# Docker Download and installation:

Open command prompt: type wsl –update



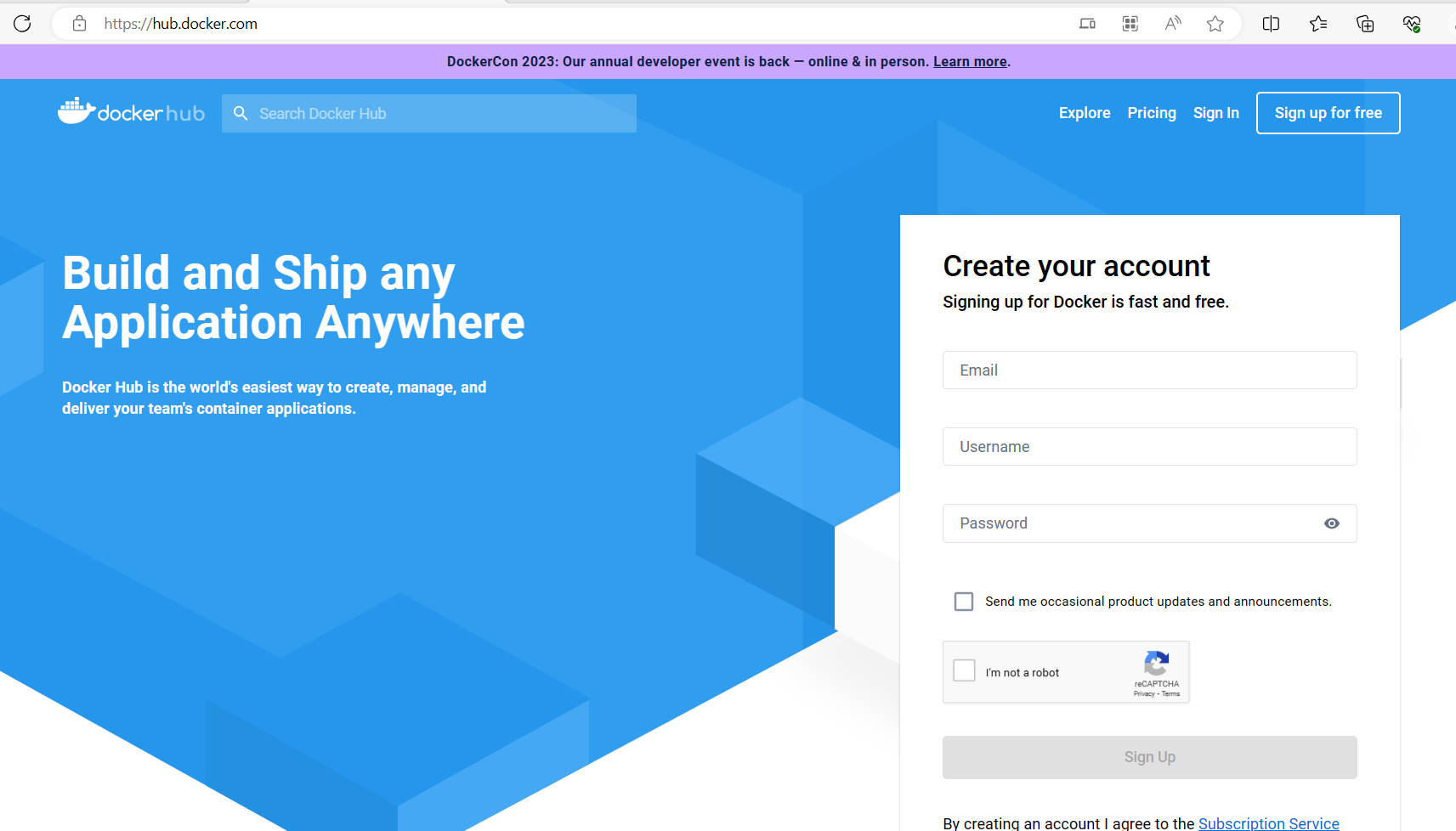
Then go to search bar and search for docker installation and go with the first link



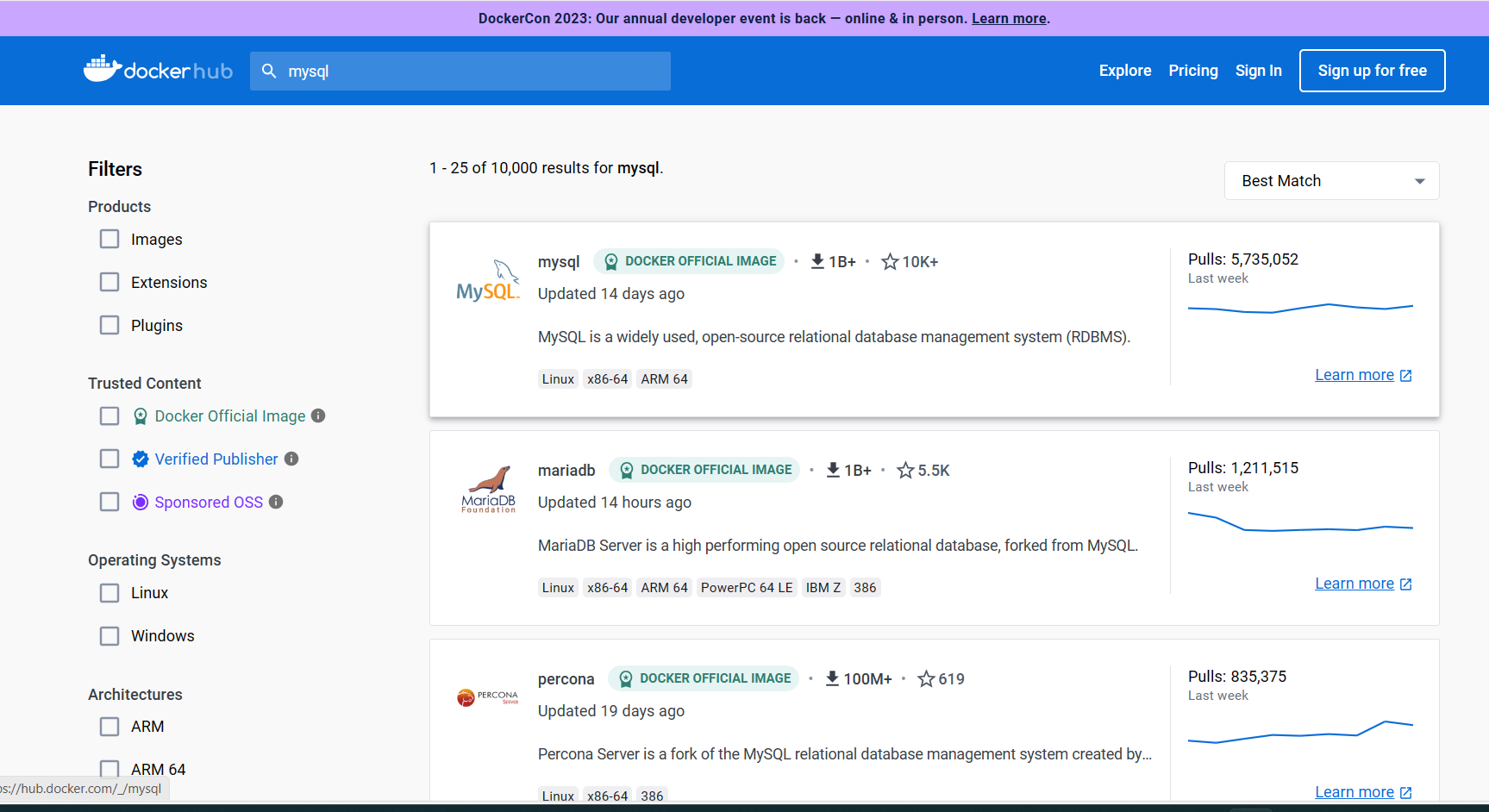
After it get downloaded it will takes few mins for installations and we can use the docker desktop on our machine.

# How to pull a docker image?

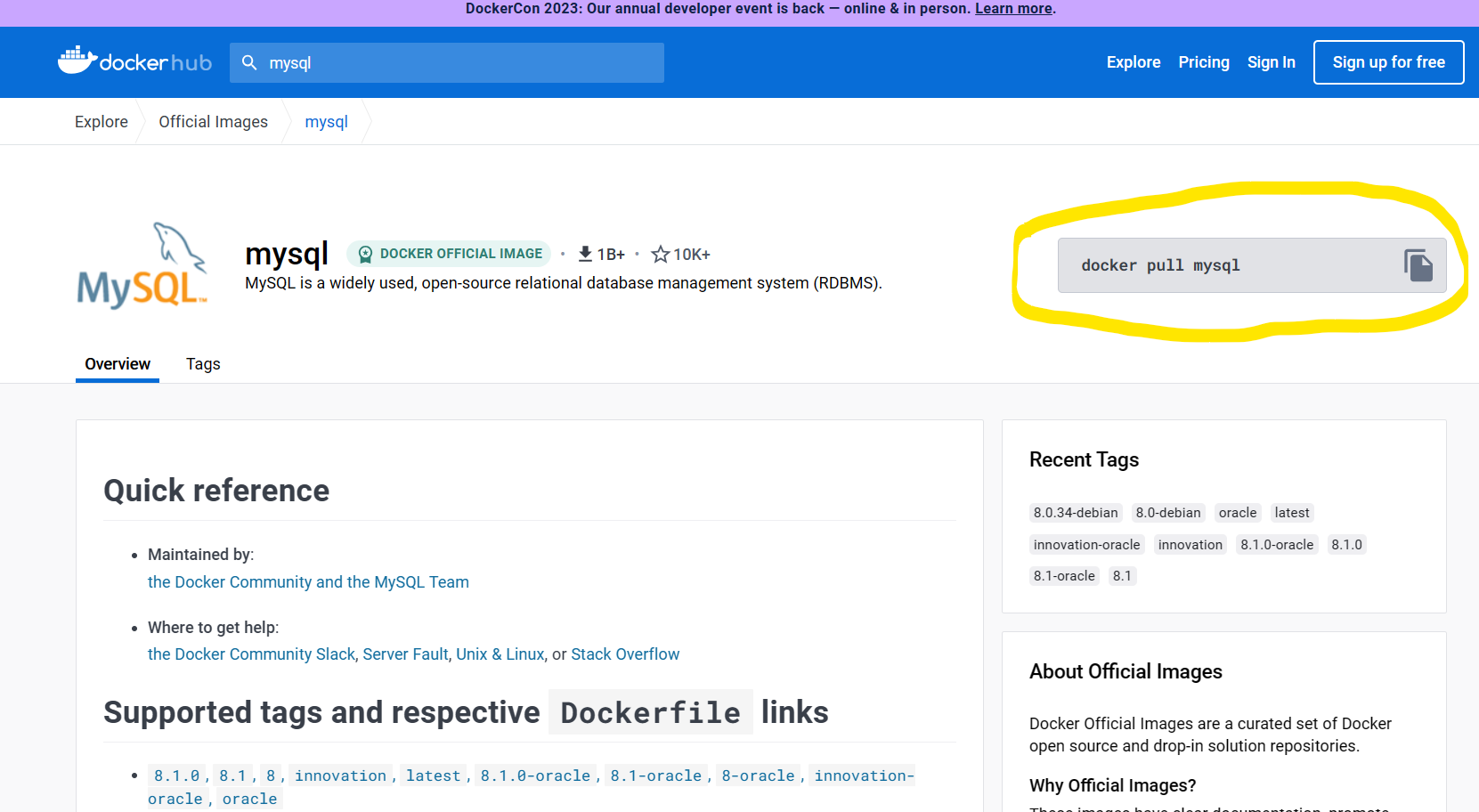
Goto hub.docker.com 🡪 [Docker Hub Container Image Library | App Containerization](https://hub.docker.com/)



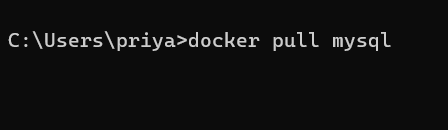
Then in the search bar search for the image we need to pull



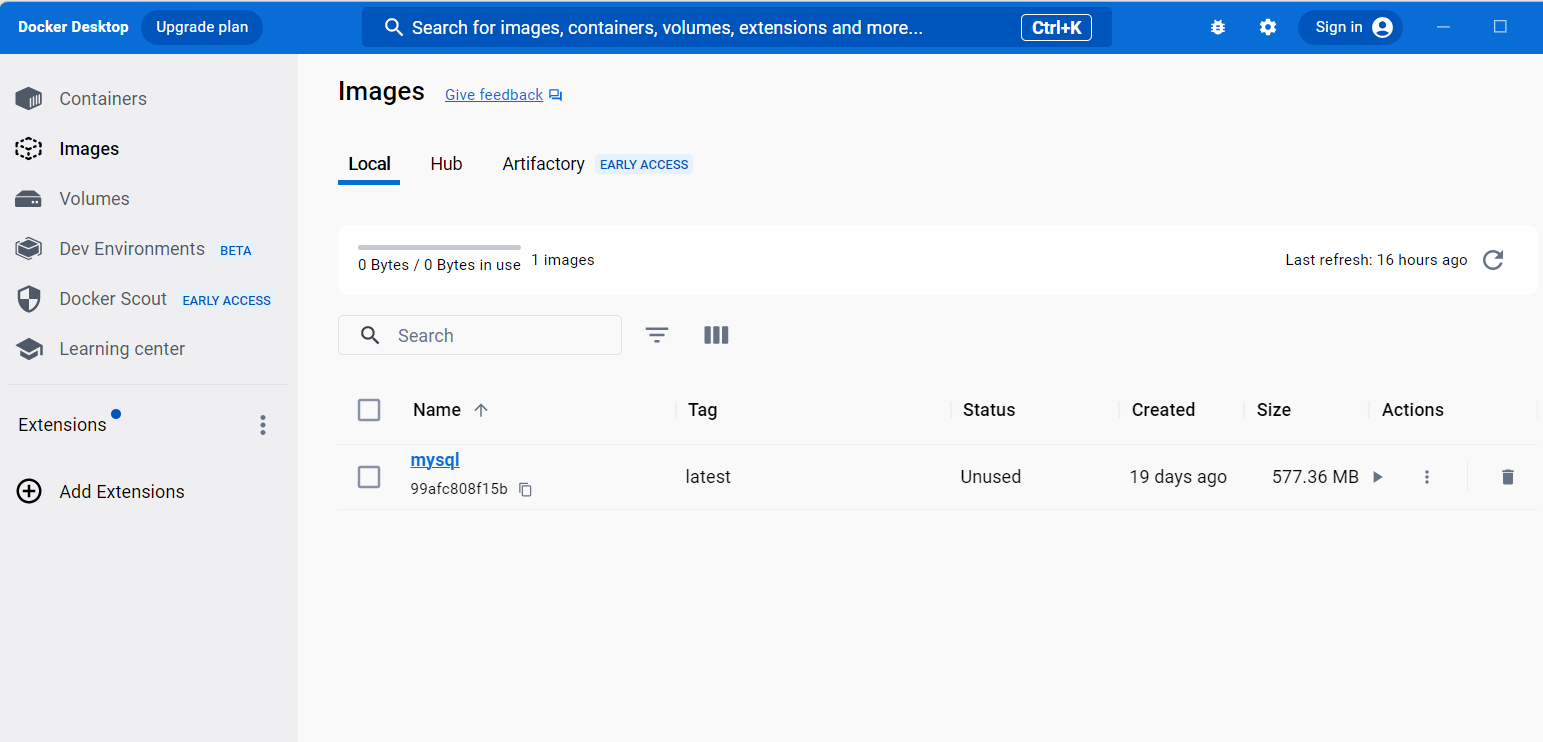
Then click on the docker image repo and we will find the command that we need to type in cmd to pull that image.



Copy that command and paste it in cmd of our machine

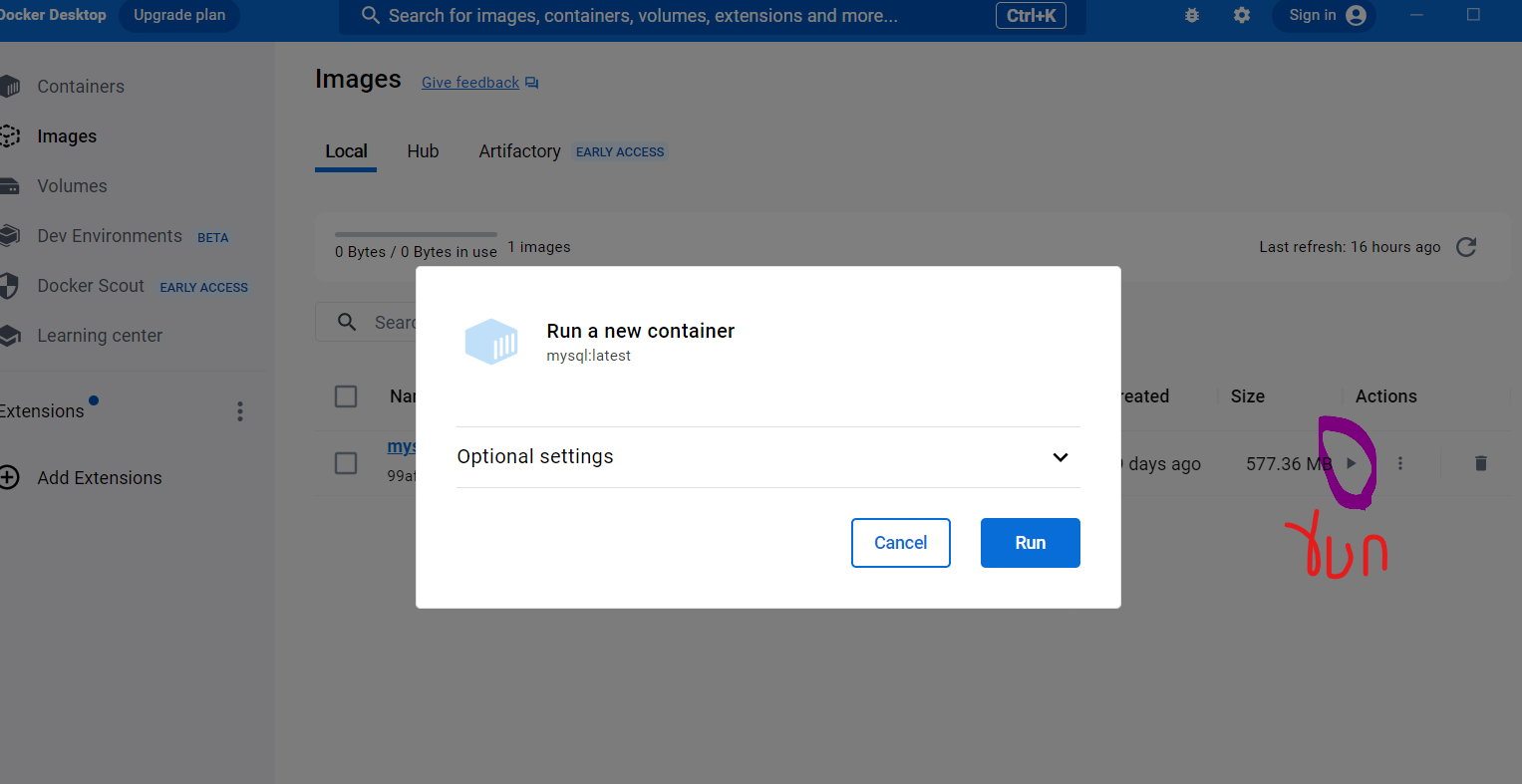


After it got pulled that image can be seen in docker desktop

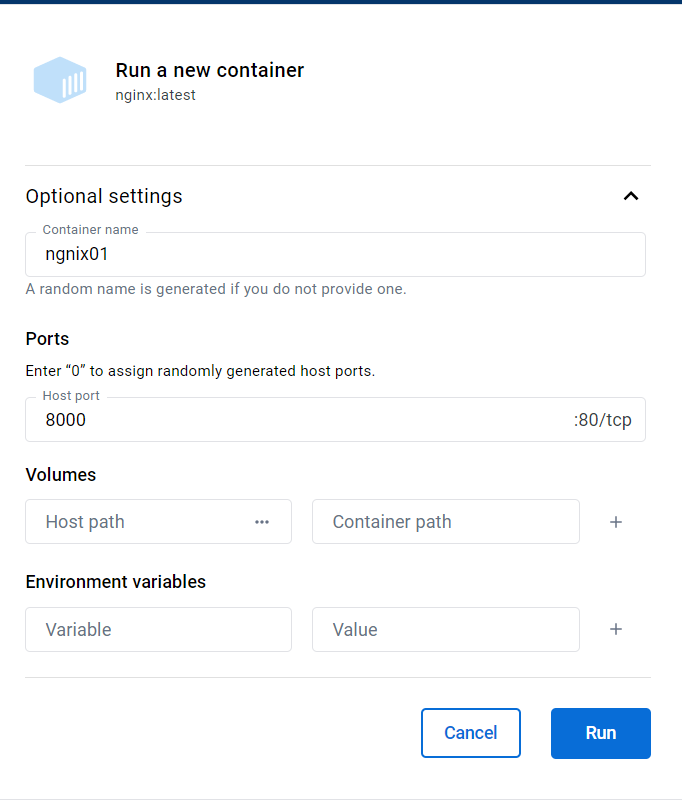


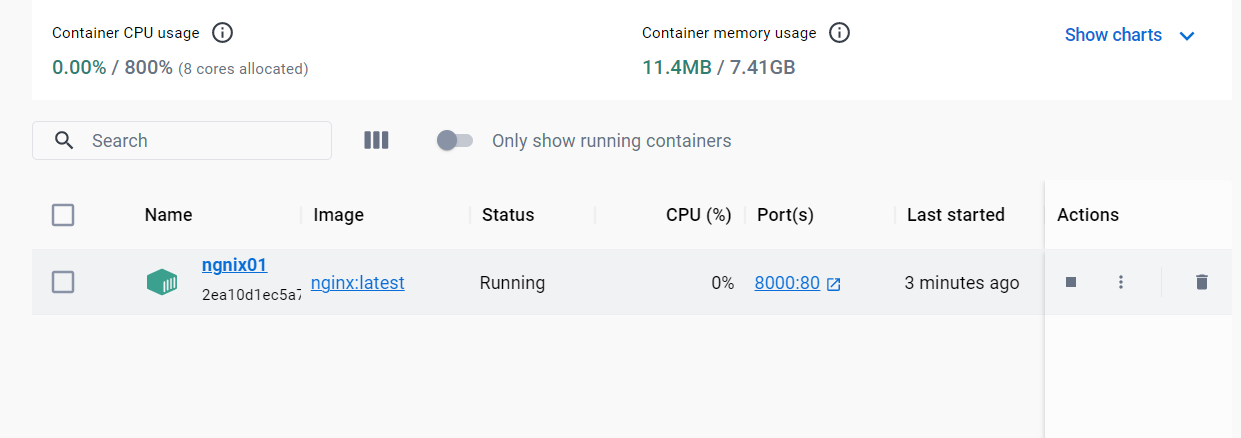
**We cannot start a container without an image. Here are the steps to create a container:**

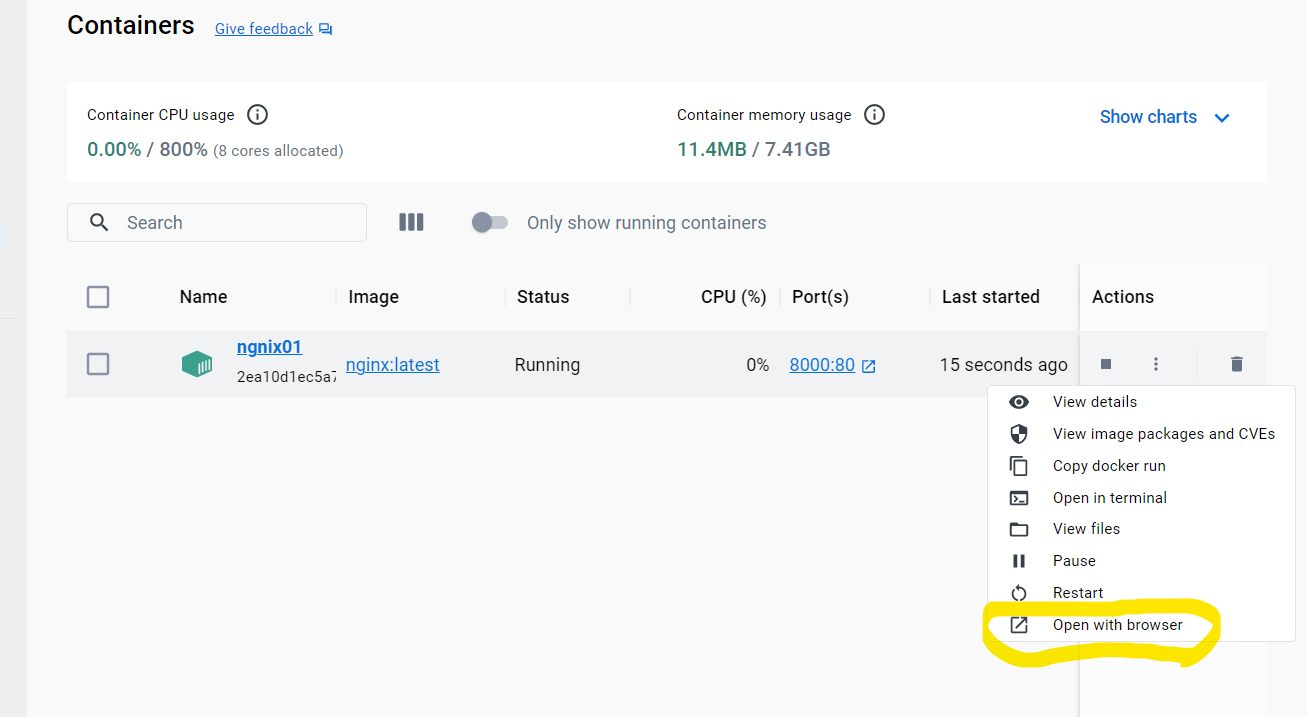
* After the image is pulled, go to docker desktop and click run on the image



* If we want to check the container in external browser then we need to add host port in optional settings by adding a name to the container we going to create



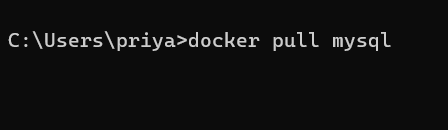
* After clicking on run the container will be created, and the terminal will be ready to use.
* In order to open the container in browser:



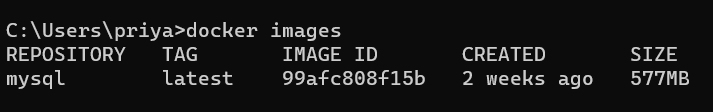
# Docker commands:

## Images:

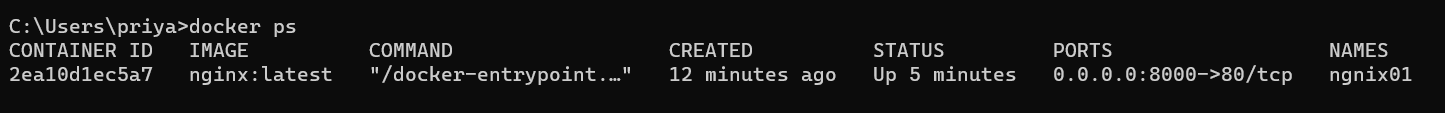
* To pull the image: **docker pull image name**



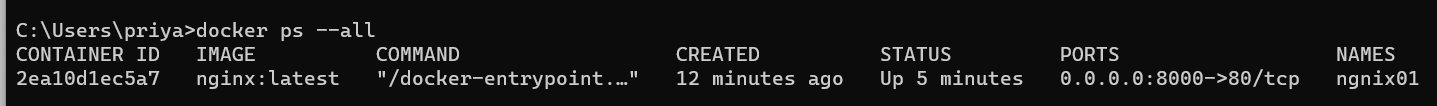
* To list local images: **docker images**



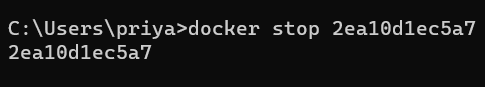
* To list currently running containers: **docker ps**

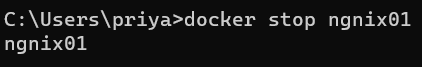


* To list all the containers (running & stopped): **docker ps –all**

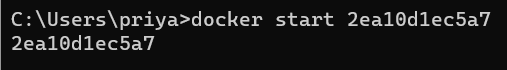


* To stop the container: **docker stop container name/ container id**





* To start the container: **docker start container name/ container id**





## **Docker Networking:**

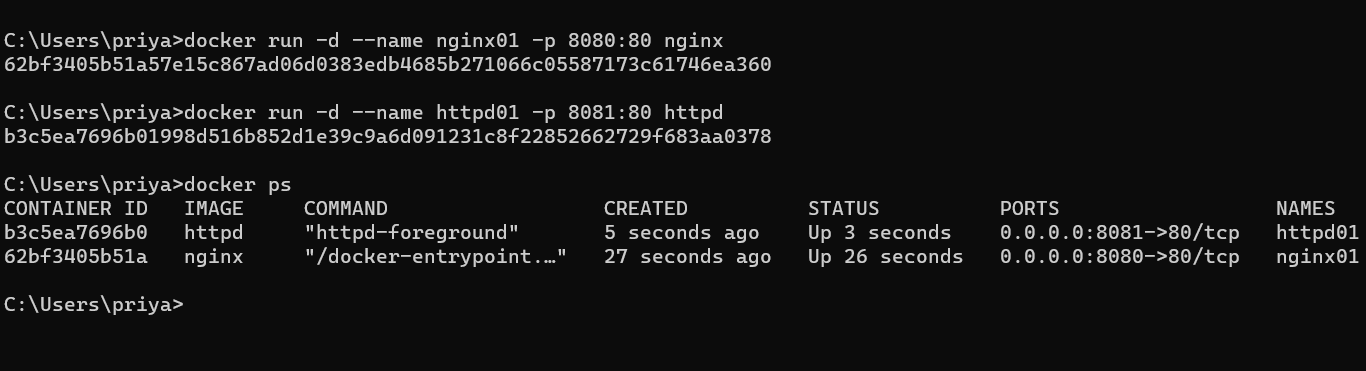
There will be three networking hosts:

* Bridge: By default, docker use this network to create containers
* Host: We need to add it explicitly
* None: Disables networking from a container

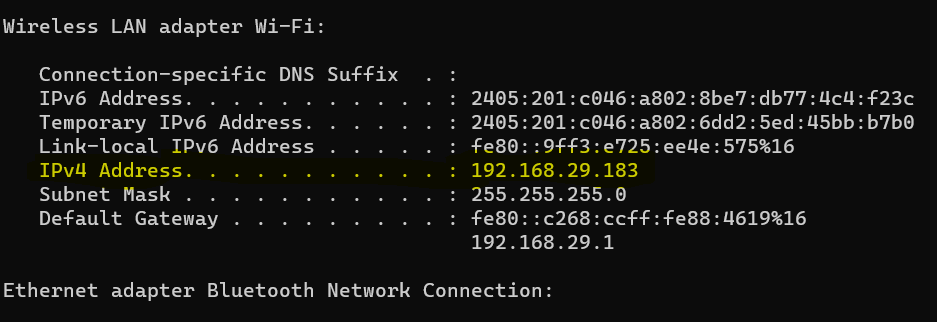
### **Bridge:**

**Whenever a docker container is created, each container use the same IP address and we can differentiate it by giving diff ports.**

**Example:** I created two docker containers named httpd01 and nginx01 by giving two diff port address.

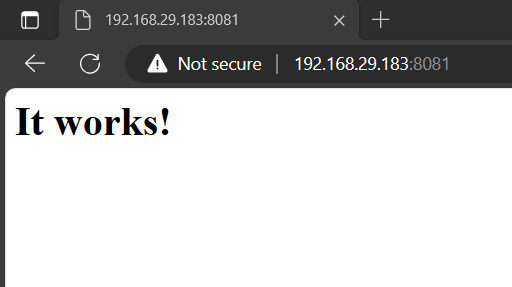


Now in order to see the IP address of the vm: use **ipconfig** command in cmd

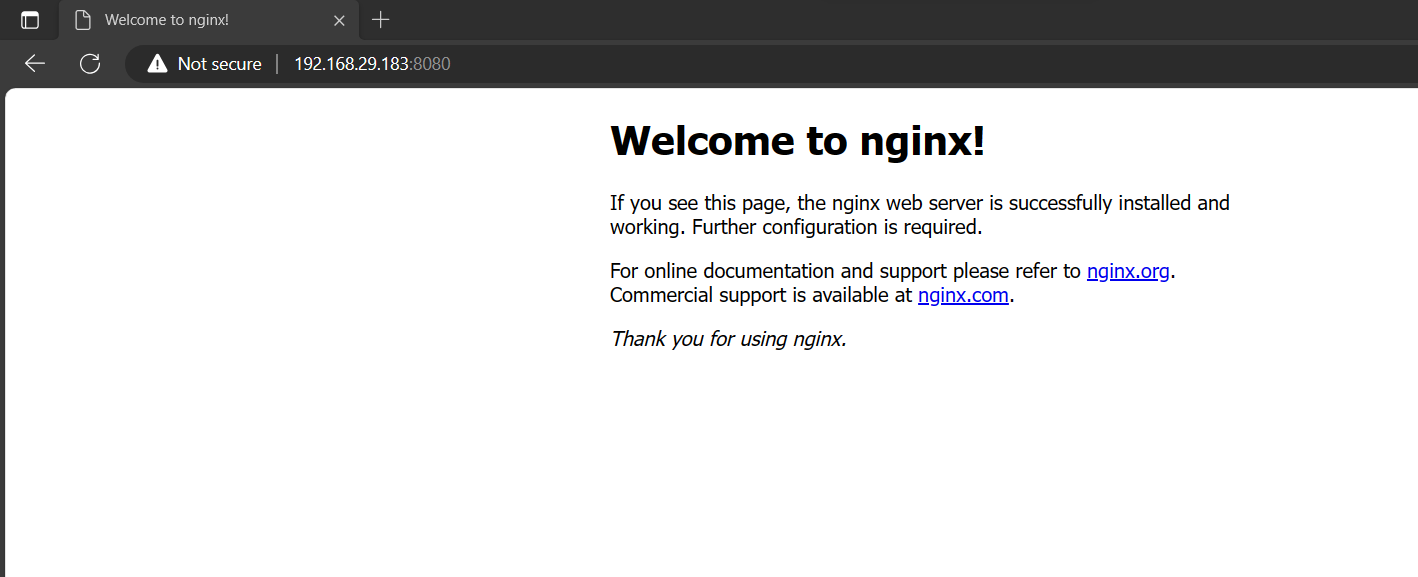


In order to check this in external terminal use the ip address along with host

1. For the container httpd01: 192.168.29.183:8081



1. For the container nginx01: 192.168.29.183:8080

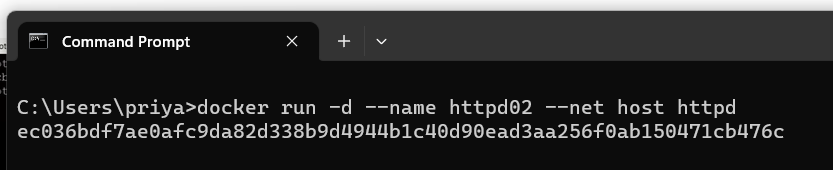


**So, if we observe here both the containers using the same IP address with diff ports.**

### **Host:**

So, if we create the container using the host we can use the IP address only for that container, we cannot use this for another container simultaneously.

**Example:** Creating the container httpd02 using the host command



Here, we no need to give the port number and if we see this in external terminal: