**Docker**

Docker permet de créer des conteneurs applicatifs à partir des images.

**Pour créer un conteneur :**

docker container run --publish <port:port> -d --name <nom\_conteneur> <image>

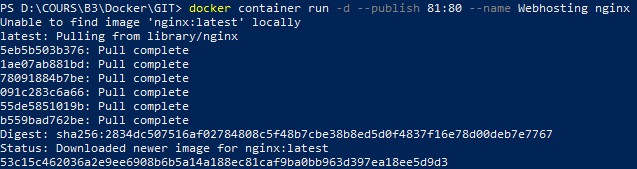
container run : permet de créer le conteneur

--publish : permet de déterminer le port utilisé par le conteneur

-d ou –detach : permet de détacher l’invite de commande du conteneur, pour ne pas le figer

--name : permet de nommer le conteneur

<image> : l’image utilisée par le conteneur. Si elle n’est pas contenue en local, elle va être téléchargée



Ici mon container a bien été créé. On peut voir le status des conteneurs avec la commande suivante :

docker ps



**Pour arrêter un conteneur :**

Docker stop <Id ou nom>

Exemple : docker stop 53





**Pour lancer un conteneur :**

docker start <Id ou nom>

Exemple : docker start 53





**Pour supprimer un conteneur :**

docker rm <ID ou nom>

Le conteneur doit être stoppé au préalable

Exemple : docker rm 53





**Pour afficher les images stockées localement :**

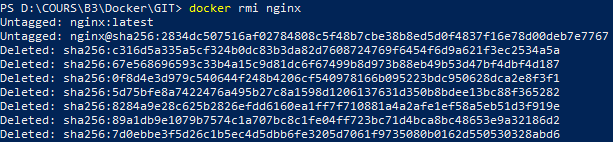
docker images



**Pour supprimer une image :**

docker rmi <ID ou nom>

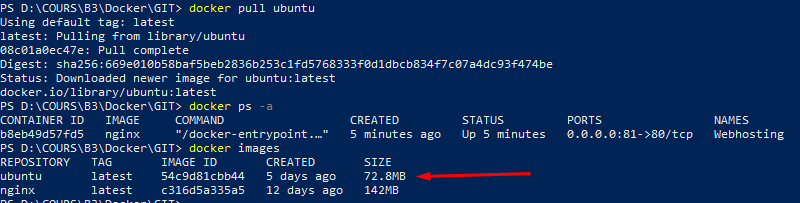
Exemple : docker rmi nginx



**Pour télécharger une image sans installer de conteneur :**

docker pull <image>

Exemple : docker pull ubuntu

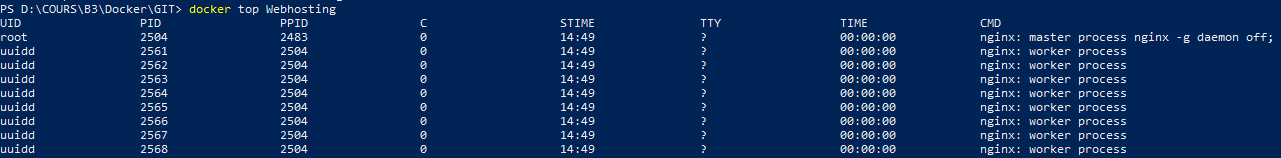


L’image est bien téléchargée mais le conteneur n’est pas créé.

**Commande TOP :**

docker top <ID ou Nom>

Exemple : docker top Webhosting



Permet d’afficher les processus actifs pour ce conteneur

**Argument IT :**

L’argument IT permet de lancer une invite de commande dans notre conteneur

Exemple : docker container run -d debian --bash