

Docker - Utiles

Formateur : Nicolas Talle - <https://github.com/Nicolab>

Pour simplifier la lecture, lorsqu'il est mention de docker *commande*, il s'agit d'une sous-commande de la commande **docker**.

- <https://docs.docker.com/get-started/>
- <https://docs.docker.com/reference/>
- <https://docs.docker.com/glossary/>

Lister les containers

La commande **ps** permet de lister les containers mémoire.

```
docker ps
```

- <https://docs.docker.com/engine/reference/commandline/ps/>

Un container est une instance d'une image.

Lister les images disponible sur le système

La commande **images** permet de lister les images précédemment téléchargées ou *buildées* (construites) sur le système.

```
docker images
```

Une image est un instantané (un snapshot) d'un système et ce qu'il contient.

Trouver des images

- <https://hub.docker.com>

Pour une recherche rapide :

```
docker search [OPTIONS] TERM
```

Exemple :

```
docker search ubuntu
```

Récupérer une image sur le registry

La commande `pull` récupère (télécharge) une image sur le système.

```
docker pull ubuntu
```

Lancer une commande dans un container

- <https://docs.docker.com/engine/reference/run/>
- <https://docs.docker.com/engine/reference/exec/>

- Lancer une commande dans un **nouveau container**.

```
docker run ubuntu:latest /bin/echo 'Hello World!'
```

Il est possible de s'y connecter.

```
docker run -it ubuntu:latest /bin/bash
```

Il est également possible de lancer un container en mode daemon `-d`.

```
docker run -d ubuntu:latest /bin/bash -c "while true; do echo $(date); sleep 1; done"
```

- Lancer une commande dans un **container en cours d'exécution**.

Exemple, lancer un nouveau container (nommé) :

```
docker run --name my_app -d ubuntu:latest /bin/bash -c "while true; do echo $(date); sleep 1; done"
```

Puis rentrer dans le container *my_app* via un autre terminal:

```
docker exec -it my_app /bin/bash
```

Afficher les logs d'un container

Afficher les 10 derniers logs:

```
docker logs my_app --tail 10
```

Afficher et streamer les 10 derniers logs:

```
docker logs my_app --tail 10 --follow
```

Monter un volume

Les volumes permettent de partager des fichiers entre le container et le système hôte (votre PC).

```
docker run -v $(pwd)/mon_fichier:/mnt/mon_fichier -it ubuntu bash
```

Démarrer un container connu

La commande `start`, démarre un container en mémoire (cf: `docker ps`).

```
docker start my_app
```

Stopper un container connu

La commande `stop`, stoppe un container en cours d'execution.

```
docker stop my_app
```

Supprimer un container

```
docker rm my_app
```

Supprimer une image

```
docker rmi ubuntu:latest
```

Supprimer les données non utilisées

Supprimer les données qui ne sont pas en cours d'utilisation.

```
docker system prune -a
```

- https://docs.docker.com/engine/reference/commandline/system_prune/

Construire (build) une image

La commande **build** permet de construire une image.

```
# Build à partir du Dockerfile dans le dossier courant  
docker build .
```

- <https://docs.docker.com/engine/reference/builder/>