

## Étape 1 – L'installation de Docker

- 1) se mettre en root puis faire :

```
apt-get update
apt-get install ca-certificates
curl -fsSL https://download.docker.com/linux/debian/gpg | gpg --dearmor -o
/etc/apt/keyrings/docker.gpg
chmod a+r /etc/apt/keyrings/docker.gpg
echo \
"deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg]
https://download.docker.com/linux/debian \
"${. /etc/os-release && echo "$VERSION_CODENAME")" stable" | \
tee /etc/apt/sources.list.d/docker.list > /dev/null
apt-get update
```

- 2) apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin  
docker-compose-plugin

- 3) docker run hello-world

Normalement un message s'affiche pour dire que l'installation s'est bien déroulée.

## Étape 2 – Découverte des commandes de base de Docker

- 1) docker pull [nom\_image] exemple : docker pull nginx

- 2) docker image ls

- 3) docker run -d -p 80:80 [nom\_image] exemple: docker run -d -p 80:80 nginx  
Pour vérifier mettre l'adresse ip utilisée dans le navigateur.

- 4) docker container ls  
nom : beautiful\_sammet  
id : e541b94b4dc5

- 5) docker ps  
status = Up depuis ... minutes si allumé

- 6) Linux e541b94b4dc5 6.1.0-11-amd64 #1 SMP PREEMPT\_DYNAMIC Debian 6.1.38-4 (2023-08-08) x86\_64 GNU/Linux
- 7) docker stop [ nom\_conteneur] exemple : docker stop beautiful\_sammet  
sortie nom\_conteneur
- 8) docker ps plus le conteneur donc pas de status.
- 9) docker restart [nom\_conteneur\_stoppé] exemple docker restart beautiful\_sammet  
sortie nom\_conteneur\_stoppé
- 10) docker ps status pareil que la 5 = up
- 11) docker rm -f [ nom\_conteneur] exemple : docker rm -f beautiful\_sammet  
sortie nom\_conteneur
- 12) docker ps
- 13) docker image rm -f [nom\_image] exemple : docker image rm -f nginx  
sortie : deleted à la dernière ligne

## Étape 3 – Création de vos propres images Docker

- 1)
 

Créer un dossier	exemple : mkdir mon_projet_docker
se déplacer dans le dossier	exemple : cd mon_projet_docker
créer un fichier dockerfile	exemple : touche Dockerfile
mettre dans le fichier les infos demandés	exemple : nano Dockerfile
# Utilisez Debian comme image de base	
FROM debian:latest si pas utilisation de debian remplacé le debian par autre chose	
# Commande qui sera exécutée lorsque le conteneur sera démarré	
CMD ["echo", "Hello World!"]	
créer l'image	exemple : docker build -t my-hello-world
- < Dockerfile	
lancer l'image	exemple : docker run my-hello-world
- 2) touch index.html pour créer un fichier index.html
- 3) nano index.html et mettre les lignes html dedans
- 4) ls et normalement le fichier index.html est présent
- 5) pas réussi

- 6) Pour lancer un conteneur faire `docker run -d -p 80:80 [nom_conteneur]`
- 7) Pour afficher le conteneur mettre dans la barre de recherche internet l'ip de l'ordinateur utilisé

## Étape 4 – Automatisation du déploiement de vos conteneurs

- 1) Pour créer un fichier de config docker compose:  
créer un fichier `docker-compose.yml`  
`touch docker-compose.yml`  
mettre dans ce fichier cela :  
version: '3'  
services:  
    my-nginx:  
        image:nginx  
        ports:  
            - "8080:80"
- 2) Pour récupérer des images :  
`docker pull [nom_image]`
- 3) `docker run -p 8080:80 [nom_image]` premier plan  
aller sur internet et mettre l'ip de l'ordi avec le port 8080
- 4) `docker stop [ nom_conteneur]`
- 5) `docker run -p 8080:80 -d [nom_image]` arrière plan
- 6) `docker compose logs -f [nom_image]`
- 7) `docker stop [nom_conteneur]`