

## **TD N°2 : Découverte l'environnement informatique : Conteneurs**

### Sommaire :

Etape 0 : .....	1
Etape 1 : .....	1
Etape 2 : .....	1
Etape 3 : .....	2
Etape 4 : .....	3
Etape 5 : .....	4
Etape 6 : .....	4

### Etape 0 :

### Etape 1 :

On vérifie si Docker est bien installé sinon on l'installe en suivant la documentation docker pour installer docker.

```
Last login: 11 Nov 11 11:48:31 CET 2023 on
root@debian11:~# docker --version
Docker version 20.10.5+dfsg1, build 55c4c88
root@debian11:~# _
```

### Etape 2 :

On créer un répertoire en utilisant la commande MKDIR sur linux la commande est mkdir mon-site-web

On se déplace ensuite dans le répertoire mon-site-web et on y créer un fichier soit en utilisant la commande touch index.html ou en utilisant nano index.html qui permettra de créer et modifier le fichier

pour y mettre un contenu html.

```
GNU nano 5.4 index.html
<html>
<head>
<title>test</title>
</head>
<body>
<h1>Bienvenue sur mon site web docker !</h1>
</body>
</html>
```

### Etape 3 :

- 1) Pour créer un fichier nommé DockerFile il suffit de faire « **touch DockerFile** » ou utilisé la commande « nano DockerFile » pour créer et modifié directement le fichier. Si on n'utilise pas la commande nano pour la prochaine etape il faudra utiliser nano pour modifier le fichier DockerFile.
- 2) On utilise donc nano en mettant « **nano DockerFile** » pour accéder au fichier précédemment créer si on n'a pas utilisé la commande nano. Ensuite dans le fichier DockerFile on entre cette ces instructions :  
« **FROM httpd:2.4**  
**COPY ./mon-site-web /usr/local/apache2/htdocs/ # Copie du contenu du répertoire local dans le répertoire du site web Apache** » Le début de la commande permet de prendre l'image d'apache qui provient de docker HUB on copie ensuite les fichiers pour le site web dans le répertoire de l'image docker. De plus la commande peut changer en fonction d'où le fichier est dans l'arborescence linux il faut donc enlever mon-site-web si le fichier DockerFile est dans mon-site-web il faut donc mettre au début de la commande ce chemin « **COPY ./ et ensuite le répertoire de l'image docker** »

```
GNU nano 5.4 Dockerfile
FROM httpd:2.4
COPY ./ /usr/local/apache2/htdocs/

[ Affichage automatique sur plusieurs lignes - arrêt ]
^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C EmplacementM-U Annuler
^X Quitter   ^R Lire fich.^_ Remplacer  ^U Coller   ^J Justifier ^_ Aller ligneM-E Refaire
```

#### Etape 4 :

Pour construire l'image docker il faut être dans le repertoire « mon-site-web » vous devez normalement être ici :

```
root@debian11:~# mkdir mon-site-web
root@debian11:~# cd mon-site-web
root@debian11:~/mon-site-web#
```

Si on se situe dans le repertoire root il suffit de faire « **cd /mon-site-web** »

**Une fois dans le bon repertoire il faut utiliser cette commande pour construire l'image docker a partir du fichier de configuration DockerFile créer précédemment : « **docker build -t mon-site-web .** »**

Pour voir si l'image a bien été construite il suffit de faire cette commande : « **Docker images** »

```
valid_lft forever preferred_lft forever
root@debian11:~/mon-site-web# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
mon-site-web        latest             2e124aaf5e48      19 minutes ago    168MB
httpd               2.4               a6ca7b52a415      19 hours ago      168MB
jlt/debian          bullseyes-ssh     909eef2b703f      14 months ago     649MB
jlt/debian          bullseyes-apache2 eeab0411fec0      14 months ago     617MB
user1/debian        bullseye-dossier  cebbc3ee43086     14 months ago     124MB
jlt/debian          bullseye-dossier  ae9964302036     14 months ago     124MB
jlt/debian          bullseye-apache2  4d21a51b718e     14 months ago     617MB
debian              bullseye          43d28810c1b4     14 months ago     124MB
hello-world         latest            feb5d9fea6a5     2 years ago       13.3kB
root@debian11:~/mon-site-web# _
```

### Etape 5 :

Une fois l'image construite il faut exécuter le conteneur contenant cette image avec cette commande :

« **docker run -d -p 8080:80 --name mon-conteneur mon-site-web** »

« -docker run » C'est la commande de base pour exécuter un conteneur Docker.

« -d » Cela signifie "detach" (détaché). Il indique à Docker de lancer le conteneur en arrière-plan, de sorte que le terminal soit libéré pour d'autres commandes.

« -p 8080:80 » Cela mappe le port 8080 de votre machine hôte au port 80 du conteneur.

« --name mon-conteneur » Cela attribue le nom "mon-conteneur" au conteneur.

« mon-site-web » C'est le nom de l'image Docker à partir de laquelle le conteneur sera créé.

Ensuite pour vérifier si le conteneur est bien exécuter on peut effectuer cette commande

« **docker ps -a** » pour voir tous les conteneurs lancés.

```
PORTS          NAMES
17e946168c5   mon-site-web   "httpd-foreground"   17 minutes ago   Up 17 minu
$              0.0.0.0:8080->80/tcp   mon-conteneur
```

### Etape 6 :

Pour vérifier si le conteneur à bien fonctionner il faut taper dans son navigateur l'adresse IP de sa machine suivi du port que l'on a mapper dans l'étape précédente, pour voir son adresse IP il suffit de taper ip a pour voir son adresse ip 10.0.0.12:8080.

```
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group 000
```

```
link/ether 08:00:27:3d:a7:0b brd ff:ff:ff:ff:ff:ff
```

```
inet 10.0.0.12/8 brd 10.255.255.255 scope global dynamic enp0s3
```

```
valid_lft 83782sec preferred_lft 83782sec
```

```
inet6 fe80::a00:27ff:fe3d:a70b/64 scope link
```

← ↻ Non sécurisé | 10.0.12:8080

**Bienvenue sur mon site web Docker !**

