



TP : DOCKER

Pour les utilisateurs de windows : <https://docs.docker.com/docker-for-windows/install/>

Pour les utilisateurs d'Ubuntu : <https://docs.docker.com/engine/install/ubuntu/>

Pour les utilisateurs Mac : <https://docs.docker.com/docker-for-mac/install/>

Après que vous installez Docker, vérifiez dans votre terminal qu'il est bien installé en exécutant la commande : **# docker version**

1- Création d'un conteneur :

Executer la commande : **# docker run hello-world**

```
C:\WINDOWS\system32>docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
b8dfde127a29: Pull complete
Digest: sha256:308866a43596e83578c7dfa15e27a73011bdd402185a84c5cd7f32a88b501a24
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/
```

Cette commande nous permet de créer un conteneur basique hello world. Puisqu'on n'a pas une image en local de hello-world, il l'a téléchargé depuis le repository Docker.

2- Créer un conteneur nginx :

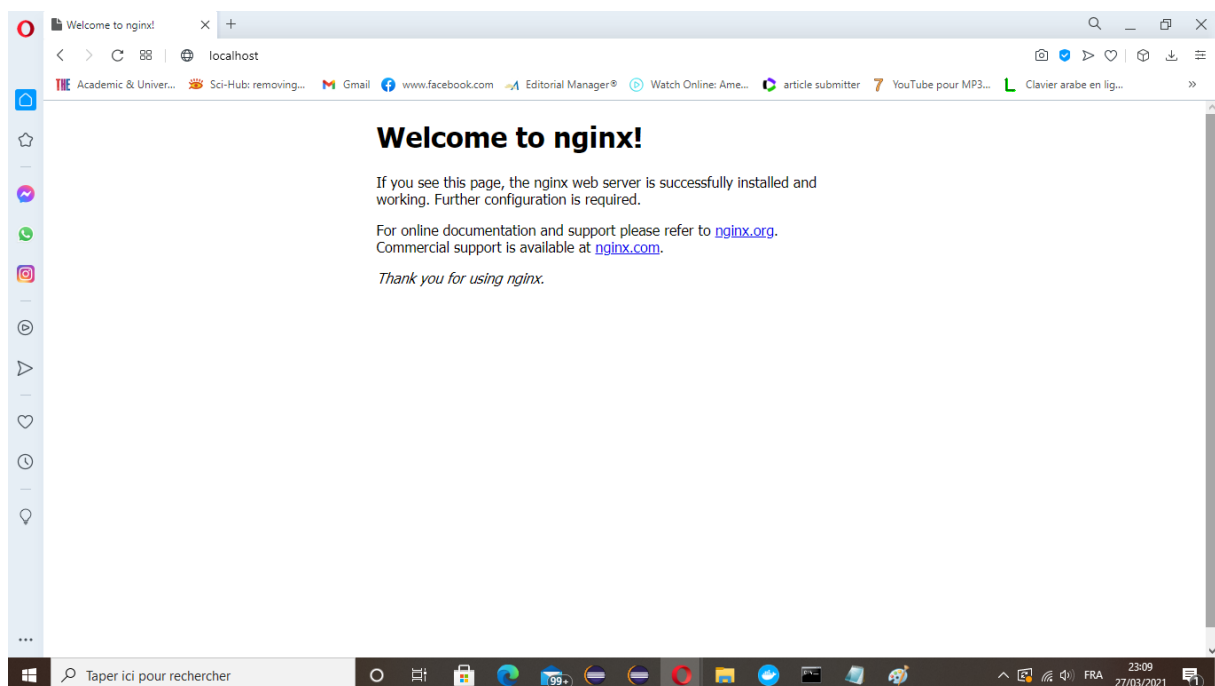
docker run --publish 80:80 nginx

```
C:\WINDOWS\system32>docker container run --publish 80:80 nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
6c2522cc7269: Pull complete
99de04de3c75: Pull complete
b0c8a51e6628: Pull complete
88b11a3d602c: Pull complete
a0e0e6bcfd2c: Pull complete
4fcb23e29ba1: Pull complete
Digest: sha256:b0ea179ab61c789ce759db491cc534e293428ad232d0df83ce44bf86261179
Status: Downloaded newer image for nginx:latest
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
172.17.0.1 - - [27/Mar/2021:21:41:02 +0000] "GET / HTTP/1.1" 200 612 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.192 Safari/537.36 OPR/74.0.3911.218" "-"
2021/03/27 21:41:03 [error] 32#32: *2 open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 172.17.0.1, server: localhost, request: "GET /favicon.ico HTTP/1.1", host: "localhost", referer: "http://localhost/"
172.17.0.1 - - [27/Mar/2021:21:41:03 +0000] "GET /favicon.ico HTTP/1.1" 404 555 "http://localhost/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.192 Safari/537.36 OPR/74.0.3911.218" "-"
172.17.0.1 - - [27/Mar/2021:21:49:12 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.192 Safari/537.36 OPR/74.0.3911.218" "-"
```

En quoi consiste le démarrage du container :

- Recherche de l'image ⇒ Si l'image n'existe pas en local, alors téléchargement via le hub.
- Configuration de l'adresse IP du container ⇒ Ainsi que de la communication entre l'extérieur et le container .
- Capture des messages entrées-sorties.

Pour accéder à Nginx il suffit de chercher localhost :80 dans votre navigateur



3. Lister les images :

#docker images -a



```
Administrateur : Invite de commandes
C:\WINDOWS\system32>
C:\WINDOWS\system32>
C:\WINDOWS\system32>docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
nginx         latest    b8cf2cbeabb9   15 hours ago   133MB
hello-world    latest    d1165f221234   3 weeks ago    13.3kB

C:\WINDOWS\system32>docker images -a
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
nginx         latest    b8cf2cbeabb9   15 hours ago   133MB
hello-world    latest    d1165f221234   3 weeks ago    13.3kB

C:\WINDOWS\system32>docker images -a -q
b8cf2cbeabb9
d1165f221234

C:\WINDOWS\system32>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
f59c5ebdbf0b   nginx     "/docker-entrypoint..." 14 minutes ago Up 14 minutes   0.0.0.0:80->80/tcp       elastic_hugle

C:\WINDOWS\system32>docker rm f59c
Error response from daemon: You cannot remove a running container f59c5ebdbf0bce242ddc8a1ae7ff9c53f75e5748596816659f115f6869b8903f. Stop the container before attempting removal or force remove

C:\WINDOWS\system32>docker rm -f f59c
f59c

C:\WINDOWS\system32>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES

C:\WINDOWS\system32>docker run hello-world
Unable to find image 'hello-world:latest' locally
docker: Error response from daemon: pull access denied for hello-world, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.

C:\WINDOWS\system32>
C:\WINDOWS\system32>docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

C:\WINDOWS\system32>docker start charming_fermat
charming_fermat

C:\WINDOWS\system32>
```

4. Lister les containers:

#docker ps -a (Pour les conteneurs arrêtés)

docker container ls

5. Démarrer un conteneur qui existe mais qui a été arrêté

#docker start ID

docker start -i ID: lancer de manière interactive

docker restart ID

```
Administrateur : Invite de commandes
C:\WINDOWS\system32>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES

C:\WINDOWS\system32>docker start --help
Usage:  docker start [OPTIONS] CONTAINER [CONTAINER...]

Start one or more stopped containers

Options:
  -a, --attach          Attach STDOUT/STDERR and forward signals
  --detach-keys string  Override the key sequence for detaching a container
  -i, --interactive     Attach container's STDIN

C:\WINDOWS\system32>docker start -i charming_fermat

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

C:\WINDOWS\system32>docker start charming_fermat
charming_fermat

C:\WINDOWS\system32>
```

6. Arrêter un container

docker container stop ID_container

```
Administrateur : Invite de commandes

3. The Docker daemon created a new container from that image which runs the
executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

C:\WINDOWS\system32>docker run --publish 80:80 nginx
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
172.17.0.1 - - [27/Mar/2021:22:09:26 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.432
4.192 Safari/537.36 OPR/74.0.3911.218" "-"

C:\WINDOWS\system32>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
07fa990dd5b2   nginx    "/docker-entrypoint..."   About a minute ago   Up About a minute   0.0.0.0:80->80/tcp        flamboyant_jang

C:\WINDOWS\system32>docker container stop 07fa99
07fa99

C:\WINDOWS\system32>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
07fa990dd5b2   nginx    "/docker-entrypoint..."   2 minutes ago   Exited (0) 16 seconds ago   0.0.0.0:80->80/tcp        flamboyant_jang
8098de4dcffa   hello-world  "/hello"                12 minutes ago   Exited (0) 12 minutes ago   0.0.0.0:80->80/tcp        charming_fermat
75843cf29b39   hello-world  "/hello"                38 minutes ago   Exited (0) 38 minutes ago   0.0.0.0:80->80/tcp        musing_bardeen

C:\WINDOWS\system32>
```

7. Supprimer un container

docker rm ID (il faut l'arrêter avant)

docker rm -f ID (forcer)

8. Afficher les logs d'un container:

#docker container logs ID

Exemple

1. Démarrer un container Mysql:

docker container run --name dbase -e MYSQL_RANDOM_ROOT_PASSWORD=yes -p 3306:3306 mysql



```
Administrateur : Invite de commandes

C:\WINDOWS\system32>docker container run --name dbase -e MYSQL_RANDOM_ROOT_PASSWORD=yes -p 3306:3306 mysql
2021-03-27 22:55:10+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.23-1debian10 started.
2021-03-27 22:55:10+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
2021-03-27 22:55:11+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 8.0.23-1debian10 started.
2021-03-27 22:55:11+00:00 [Note] [Entrypoint]: Initializing database files
2021-03-27 22:55:11.100315Z 0 [System] [MY-013169] [Server] /usr/sbin/mysqld (mysqld 8.0.23) initializing of server in progress as process 42
2021-03-27 22:55:11.116744Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2021-03-27 22:55:21.898033Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2021-03-27 22:55:51.795535Z 6 [Warning] [MY-010453] [Server] root@localhost is created with an empty password ! Please consider switching off the --initialize-insecure option.
2021-03-27 22:56:45+00:00 [Note] [Entrypoint]: Database files initialized
2021-03-27 22:56:45+00:00 [Note] [Entrypoint]: Starting temporary server
2021-03-27 22:56:46.062940Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.0.23) starting as process 87
2021-03-27 22:56:46.181809Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2021-03-27 22:56:47.273299Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2021-03-27 22:56:47.615077Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Socket: /var/run/mysqld/mysqld.sock
2021-03-27 22:56:48.314813Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.
2021-03-27 22:56:48.315274Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections are now supported for this channel.
2021-03-27 22:56:48.410873Z 0 [Warning] [MY-011810] [Server] Insecure configuration for --pid-file: Location '/var/run/mysqld' in the path is accessible to all OS users. Consider choosing a different directory.
2021-03-27 22:56:48.442347Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.0.23' socket: '/var/run/mysqld/mysqld.sock' port: 0 MySQL Community Server - GPL.
2021-03-27 22:56:48+00:00 [Note] [Entrypoint]: Temporary server started.
Warning: Unable to load '/usr/share/zoneinfo/iso3166.tab' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/leap-seconds.list' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/zone.tab' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/zone1970.tab' as time zone. Skipping it.
2021-03-27 22:56:57+00:00 [Note] [Entrypoint]: GENERATED ROOT PASSWORD: teihuasaeXunu0eithieX4laeTee0xie
2021-03-27 22:56:57+00:00 [Note] [Entrypoint]: Stopping temporary server
2021-03-27 22:56:57.454961Z 10 [System] [MY-013172] [Server] Received SHUTDOWN from user root. Shutting down mysqld (Version: 8.0.23).
2021-03-27 22:57:13.832892Z 0 [System] [MY-010910] [Server] /usr/sbin/mysqld: Shutdown complete (mysqld 8.0.23) MySQL Community Server - GPL.
2021-03-27 22:57:14+00:00 [Note] [Entrypoint]: Temporary server stopped

2021-03-27 22:57:14+00:00 [Note] [Entrypoint]: MySQL init process done. Ready for start up.

2021-03-27 22:57:14.780784Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.0.23) starting as process 1
2021-03-27 22:57:14.795613Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2021-03-27 22:57:16.174075Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2021-03-27 22:57:16.512643Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Bind-address: '::' port: 33060, socket: /var/run/mysqld/mysqld.sock
2021-03-27 22:57:16.992791Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.
2021-03-27 22:57:16.993267Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections are now supported for this channel.
2021-03-27 22:57:17.050802Z 0 [Warning] [MY-011810] [Server] Insecure configuration for --pid-file: Location '/var/run/mysqld' in the path is accessible to all OS users
```

2. effectuez des commandes sur le container (Mysql)

docker exec -it dbase mysql -uroot -p (i for interactive, t for terminal)

```
Administrateur : Invite de commandes - docker exec -it dbase mysql -uroot -p

Options:
  -d, --detach          Detached mode: run command in the background
  --detach-keys string  Override the key sequence for detaching a
                        container
  -e, --env list         Set environment variables
  --env-file list       Read in a file of environment variables
  -i, --interactive      Keep STDIN open even if not attached
  --privileged          Give extended privileges to the command
  -t, --tty             Allocate a pseudo-TTY
  -u, --user string      Username or UID (format:
                        <name|uid[:<group|gid>])
  -w, --workdir string   Working directory inside the container

C:\WINDOWS\system32>docker exec -it dbase mysql -uroot -p
Enter password:
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)

C:\WINDOWS\system32>docker exec -it dbase mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.23 MySQL Community Server - GPL

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.00 sec)

mysql>
```

3. L'intérêt c'est d'enregistrer vos modification et rendre votre container en image pour qu'il soit prêt à utiliser. Pour cela on va créer un compte sur dockerhub et pusher notre image dans le répertoire



#docker commit -m 'message' ID new_image_name : Pour convertir le container en image

```
C:\WINDOWS\system32>docker commit --help

Usage:  docker commit [OPTIONS] CONTAINER [REPOSITORY[:TAG]]

Create a new image from a container's changes

Options:
  -a, --author string      Author (e.g., "John Hannibal Smith
                           <hannibal@a-team.com>")
  -c, --change list        Apply Dockerfile instruction to the created image
  -m, --message string     Commit message
  -p, --pause              Pause container during commit (default true)

C:\WINDOWS\system32>docker commit -m "initial commit" dbase my_table
sha256:ddc0f77c4066fa3bf7b0f8f8fabe156e24fe9461e3bf88e6dbadf26782b256ba
```

#docker login --username=... : Pour se connecter sur DockerHub

docker tag image_name name_repos : Pour donner un tag à votre image

docker push name_repos:latest : Pour le publier dans votre répertoire

```
C:\WINDOWS\system32>docker login --username=brookkk
Password:
2eb61bab0d29: Pushing [=====>] 363.7MB/410.8MB
2eb61bab0d29: Pushing [=====>] 359.2MB/410.8MB
C:\WINDOWS\system32>docker login --username=brookkk
Password:
Error response from daemon: Get https://registry-1.docker.io/v2/: unauthorized: incorrect username or password

C:\WINDOWS\system32>docker login --username=brookkk
Password:
Login Succeeded

2eb61bab0d29: Pushing [=====>] 357.6MB/410.8MB
Using default tag: latest
The push refers to repository [docker.io/library/my_table]
c13c872bab0b: Preparing
845774328672: Preparing
49f85bcc958a: Preparing
058dd643edd4: Preparing
2eb61bab0d29: Preparing
121f3117307b: Waiting
```