Basile LE THIEC

TP - R409

Exercice 1 :

1)
```
docker run debian:10.13
```

2)
```
docker run debian:latest du -h /etc
```

3)
```
docker run -it debian:latest
```

Exercice 2 :

1)
```
docker run alpine:latest sleep infinity

docker container list

docker exec <ct_name> hostname
```

```
~/Desktop/R409 main*
❯ docker exec pedantic_satoshi hostname
7cfc82e4cd61
```

Il est identique à l'ID du conteneur, donc *docker ps -a* serait suffisant.

```
~/Desktop/R409 main*
❯ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED         STATUS         PORTS                    NAMES
7cfc82e4cd61   alpine:latest  "sleep infinity"         9 minutes ago   Up 9 minutes                            pedantic_satoshi
```

2)
```
docker run alpine:latest sleep infinity
```

Puis dans un autre terminal
```
docker exec <ct_name> <cmd>
```

3)
```
docker ps -a
```

Basile LE THIEC

Exercice 4 :

```
❯ docker build -t ex4 .
[+] Building 0.0s (8/8) FINISHED                                                                    docker:desktop-
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 342B
 => [internal] load metadata for docker.io/library/nginx:latest
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [internal] load build context
 => => transferring context: 33B
 => [1/3] FROM docker.io/library/nginx:latest
 => CACHED [2/3] COPY ./fichier.zip /usr/share/nginx/html/
 => CACHED [3/3] RUN apt update &&     apt install -y unzip &&     rm /usr/share/nginx/html/index.html &&     unzip /usr/share/nginx/html/fichier.zip
 => exporting to image
 => => exporting layers
 => => writing image sha256:70f1cd72e7439f6db4cca879c8992064da7909a46f37b8923e0e450f9933f4f8
 => => naming to docker.io/library/ex4

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/h0qkkjsckcy2d0ahrs5mp3ejp

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview
```

```
docker run -p 8080:80 ex4
```

```
❯ curl localhost:8080
<!doctype html>
<html>
    <head>
        <link href="index.css" rel=stylesheet type="text/css">
        <title> Mon titre </title>
    </head>
        <h1> Mon titre </h1>
        <p> c'est mon site web </p>
</html>
```

```
docker inspect peaceful_keller
```

```
          },
          "HairpinMode": false,
          "LinkLocalIPv6Address": "",
          "LinkLocalIPv6PrefixLen": 0,
          "SecondaryIPAddresses": null,
          "SecondaryIPv6Addresses": null,
          "EndpointID": "7c089fd3836a8f636eeab94ec5d24cff1099a6746eed11aa4340ba642610d3d9",
          "Gateway": "172.17.0.1",
          "GlobalIPv6Address": "",
          "GlobalIPv6PrefixLen": 0,
          "IPAddress": "172.17.0.2",
          "IPPrefixLen": 16,
          "IPv6Gateway": "",
          "MacAddress": "02:42:ac:11:00:02",
          "Networks": {
              "bridge": {
                  "IPAMConfig": null,
                  "Links": null,
                  "Aliases": null,
                  "MacAddress": "02:42:ac:11:00:02",
                  "NetworkID": "0be59016fd6a1c2b613e889adef1aa16510f881dd13735e3a8a2bdf3ef5b1d89",
                  "EndpointID": "7c089fd3836a8f636eeab94ec5d24cff1099a6746eed11aa4340ba642610d3d9",
                  "Gateway": "172.17.0.1",
                  "IPAddress": "172.17.0.2",
```

```
  ~/Desktop/R409/TP-test/ex4 main*
⊗ ❯ curl 172.17.0.2
  ^C
```

Avec Docker Mac OS il n’est pas possible de communiquer de la machine hôte vers un conteneur Docker sans exposer de port au préalable et en utilisant les paramètres réseaux Docker par défaut.

```
docker inspect peaceful_keller
```


```
..
"Mounts": [
    {
        "Type": "volume",
        "Name": "13f8c9046ac4dc7f36b5999563606102afe23537ed145f2aa09f6b76159f0a76",
        "Source": "/var/lib/docker/volumes/13f8c9046ac4dc7f36b5999563606102afe23537ed145f2aa09f6b76159f0a76/_data",
        "Destination": "/usr/share/nginx/html",
        "Driver": "local",
        "Mode": "",
        "RW": true,
        "Propagation": ""
    }
```