Các lệnh cơ bản thao tác với Docker

1 docker --version

```
C:\Users\Student>docker --version
Docker version 24.0.6, build ed223bc
```

2 docker run hello-world

```
C:\Users\Student>docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
e6590344b1a5: Pull complete
Digest: sha256:7e1a4e2d11e2ac7a8c3f768d4166c2defeb09d2a750b010412b6ea13de1efb19
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.
```

3 docker pull nginx

```
C:\Users\Student>docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
5e909acdb790: Pull complete
5eaa34f5b9c2: Pull complete
617c4bccf534: Pull complete
e7e0ca015e55: Pull complete
973fe654e984: Pull complete
d97f5c0f51d43: Pull complete
c22eb46e871a: Pull complete
Digest: sha256:124b44bfc9ccd1f3cedf4b592d4d1e8bddb78b51ec2ed5056c52d3692baebc19
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest

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

4 docker images

```
C:\Users\Student>docker images
REPOSITORY    TAG       IMAGE ID       CREATED       SIZE
nginx         latest    53a18edff809   6 weeks ago   192MB
hello-world   latest    74cc54e27dc4   8 weeks ago   10.1kB
```

5 docker run -d nginx

```
C:\Users\Student>docker run -d nginx
cda100b6f5766a78299e1c54b836e7c2d17a1f02626238630ad3c859452c17cd
```

6 docker ps

```
C:\Users\Student>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS     NAMES
cda100b6f576   nginx     "/docker-entrypoint.…"   23 seconds ago   Up 20 seconds   80/tcp    trusting_mestorf
```

7 docker ps -a

```
C:\Users\Student>docker ps -a
CONTAINER ID   IMAGE         COMMAND                  CREATED          STATUS                      PORTS     NAMES
cda100b6f576   nginx         "/docker-entrypoint.…"   43 seconds ago   Up 40 seconds               80/tcp    trusting_mes
torf
98902123d9e2   hello-world   "/hello"                 3 minutes ago    Exited (0) 3 minutes ago              compassionat
e_beaver
```

8 docker logs <container_id>

```
C:\Users\Student>docker logs 98902123d9e2

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/
```

9 docker exec -it <container_id> /bin/sh

10 docker stop <container_id>

```
C:\Users\Student>docker stop 98902123d9e2
98902123d9e2
```

11 docker restart <container_id>

```
C:\Users\Student>docker restart 98902123d9e2
98902123d9e2
```

12 docker rm <container_id>

```
C:\Users\Student>docker rm cda100b6f576
cda100b6f576
```

13 docker container prune

```
C:\Users\Student>docker container prune
WARNING! This will remove all stopped containers.
Are you sure you want to continue? [y/N] y
Deleted Containers:
98902123d9e2f8b156ddb733215f74aded35d745c674d151eeb74f6d3d73ea95

Total reclaimed space: 0B
```

14 docker rmi <image_id>

```
C:\Users\Student>docker rmi 53a18edff809
Untagged: nginx:latest
Untagged: nginx@sha256:124b44bfc9ccd1f3cedf4b592d4d1e8bddb78b51ec2ed5056c52d3692baebc19
Deleted: sha256:53a18edff8091d5faff1e42b4d885bc5f0f897873b0b8f0ace236cd5930819b0
Deleted: sha256:9624c14fde1debdc1256228b54278fec5e576a42dcbf73f420762a91f4a06c87
Deleted: sha256:75cef3a8c4e762e0d3d0c01fbe5cf9407478057005f945fa78edef29a2bc6e33
Deleted: sha256:bf22610f6a6c90cb4a456617b926c87cb1c50efd3f90b1d96d9c88e5f4b75a6e
Deleted: sha256:8e41d2be566aeafda18718a8a4b8c515c50b06f82cd7a92420ae91010773e15c
Deleted: sha256:da2d6794d8696a98178b6882353953c9f410dcffff428cfa3caa5759036d24bd
Deleted: sha256:e9228041e2928859e124edaf5a456926097605092e1855d51aa9e43f984f770e
Deleted: sha256:1287fbecdfcce6ee8cf2436e5b9e9d86a4648db2d91080377d499737f1b307f3
```

15 docker image prune -a

```
C:\Users\Student>docker image prune -a
WARNING! This will remove all images without at least one container associated to them.
Are you sure you want to continue? [y/N] y
Deleted Images:
untagged: hello-world:latest
untagged: hello-world@sha256:7e1a4e2d11e2ac7a8c3f768d4166c2defeb09d2a750b010412b6ea13de1efb19
deleted: sha256:74cc54e27dc41bb10dc4b2226072d469509f2f22f1a3ce74f4a59661a1d44602
deleted: sha256:63a41026379f4391a306242eb0b9f26dc3550d863b7fdbb97d899f6eb89efe72

Total reclaimed space: 10.07kB
```

16 docker run -d -p 8080:80 nginx

```
C:\Users\Student>docker run -d -p 8080:80 nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
6e909acdb790: Pull complete
5eaa34f5b9c2: Pull complete
417c4bccf534: Pull complete
e7e0ca015e55: Pull complete
373fe654e984: Pull complete
97f5c0f51d43: Pull complete
c22eb46e871a: Pull complete
Digest: sha256:124b44bfc9ccd1f3cedf4b592d4d1e8bddb78b51ec2ed5056c52d3692baebc19
Status: Downloaded newer image for nginx:latest
71d45d249049fee1e6bd2f5a6ee7a9d79dbfbe520ca1e04978a6d7f694080c11
```

17 docker inspect <container_id>

```
C:\Users\Student>docker inspect 71d45d249049
[
    {
        "Id": "71d45d249049fee1e6bd2f5a6ee7a9d79dbfbe520ca1e04978a6d7f694080c11",
        "Created": "2025-03-20T00:35:32.862553Z",
        "Path": "/docker-entrypoint.sh",
        "Args": [
            "nginx",
            "-g",
            "daemon off;"
        ],
        "State": {
            "Status": "running",
            "Running": true,
            "Paused": false,
            "Restarting": false,
            "OOMKilled": false,
            "Dead": false,
            "Pid": 3185,
            "ExitCode": 0,
            "Error": "",
            "StartedAt": "2025-03-20T00:35:45.4454353Z",
            "FinishedAt": "0001-01-01T00:00:00Z"
        },
```

18 docker run -d -v mydata:/data nginx

```
C:\Users\Student>docker run -d -v mydata:/data nginx
f50290572944e272b3ebd77763b51f8457ab65c950853372983715b835f187ae
```

19 docker volume ls

```
C:\Users\Student>docker volume ls
DRIVER    VOLUME NAME
local     mydata
```

20 docker volume prune

```
C:\Users\Student>docker volume prune
WARNING! This will remove anonymous local volumes not used by at least one container.
Are you sure you want to continue? [y/N] y
Total reclaimed space: 0B
```

21 docker run -d --name my_nginx nginx

```
C:\Users\Student>docker run -d --name my_nginx nginx
6b4b8841f73ebdeaeda8093ebf330241795129060be78c1985b4652d08631248
```

22 docker stats

```
CONTAINER ID   NAME              CPU %     MEM USAGE / LIMIT     MEM %     NET I/O           BLOCK I/O   PIDS
6b4b8841f73e   my_nginx          0.00%     10.01MiB / 5.998GiB   0.16%     726B / 0B         0B / 0B     13
f50290572944   agitated_shamir   0.00%     9.93MiB / 5.998GiB    0.16%     796B / 0B         0B / 0B     13
71d45d249049   zen_shaw          0.00%     9.996MiB / 5.998GiB   0.16%     5.56kB / 3.12kB   0B / 0B     13
```

23 docker network ls

```
C:\Users\Student>docker network ls
NETWORK ID     NAME      DRIVER    SCOPE
4de389d272f6   bridge    bridge    local
3ca57c3e562c   host      host      local
4ef757842cb0   none      null      local
```

24 docker network create my_network

```
C:\Users\Student>docker network create my_network
772199baabba74fd31bf5c577dc116cb8db027f290959b95c977461cf817a499
```

25 docker run -d --network my_network --name my_container nginx

```
C:\Users\Student>docker run -d --network my_network --name my_container nginx
c94d0ae3dd8fd7a884dd801e76dc929f8d41d01768f43e346578070a5013905d
```

26 docker network connect my_network my_nginx

```
C:\Users\Student>docker network connect my_network my_nginx
```

27 docker run -d -e MY_ENV=hello_world nginx

```
C:\Users\Student>docker run -d -e MY_ENV=hello_world nginx
7c2bd1bfa46430d27349bb69772d345bf489d7e70890167fb55400cc7f4709f6
```

28 docker logs -f my_nginx

```
C:\Users\Student>docker logs -f my_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: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/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
2025/03/20 00:38:23 [notice] 1#1: using the "epoll" event method
2025/03/20 00:38:23 [notice] 1#1: nginx/1.27.4
2025/03/20 00:38:23 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2025/03/20 00:38:23 [notice] 1#1: OS: Linux 5.10.16.3-microsoft-standard-WSL2
2025/03/20 00:38:23 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2025/03/20 00:38:23 [notice] 1#1: start worker processes
2025/03/20 00:38:23 [notice] 1#1: start worker process 29
2025/03/20 00:38:23 [notice] 1#1: start worker process 30
```

29 FROM nginx

COPY index.html /usr/share/nginx/html/index.html

30 docker build -t my_nginx_image .

```
C:\Users\Student>docker build -t my_nginx_image .
[+] Building 2.3s (2/2) FINISHED                                                docker:default
 => [internal] load .dockerignore                                                         0.9s
 => => transferring context: 2B                                                           0.0s
 => [internal] load build definition from Dockerfile                                      0.6s
 => => transferring dockerfile: 2B                                                        0.0s
```

31 docker run -d -p 8080:80 my_nginx_image