

Docker Basic

Docker naming convention in docker hub

- In docker registry, we have a repository (are just a folder to keep your images organized in the registry)
- Tags are just to identify the exact image in the repository Tag=Image name

```
<username>/repository name:tag  
leonardtia/devops-pro-repos:stutendent_registration
```

Create a repository

Delete a repository

Pulling an image from the docker hub

This will pull the latest image

```
docker pull <images name>  
docker pull centos
```

Pulling a specific image from the docker hub

- In docker hub, click on explore and search for the image that you want
- select the official image and Tags
- choose the version that you want

```
docker pull centos:6.9  
docker pull centos:6.10
```

List all the images

```
docker images
```

Docker Pull command

It will pull the image and store it locally but it will not run it. It will only pull only if the image isn't present locally.

```
docker pull <images name>  
docker pull centos
```

Docker Run command

it will check if we have the image on the local machine first. If yes, it will run. If not, it will pull the images and it will run it

```
docker run centos
```

Run the container in interactive mode and get in or root user mode

A random name will be assigned to the container by docker daemon if you do not assign your own name.

```
docker run -it <image name> bash
Or
docker run -it <image name> /bin/bash

docker run -it centos:latest bash

docker run -it <container name> <image name:tag or ID> bash
Or
docker run -it <image name> image name:tag or ID /bin/bash

docker run -it my-centos centos:latest bash
```

Run the container detach mode

To start the container and keep it running in the background, run it in detached mode. *This is good for web servers*

- **I** = interactive
- **T** = terminal
- **D** = detach or run the container in the background

```
docker run -itd <image name> bash
Or
docker run -it -d <image name> /bin/bash

docker run -itd centos:latest bash

docker run -itd <container name> <image name:tag or ID> bash
Or
docker run -it -d <image name> image name:tag or ID /bin/bash

docker run -itd my-centos centos:latest bash
```

List all the running containers

```
docker ps
```

List all active (running) and inactive (excited) containers

a = all

```
docker ps -a
```

run commands in the running containers from the docker host

```
docker exec <running container ID> <command>
docker exec a79012e25f34 cat /etc/*release
docker exec -it 4c6f8c107730 mkdir -p /tmp/tia
docker exec -it 4c6f8c107730 ls -l /tmp/
```

Login into a running container

```
docker attach <running container ID> bash
Or
docker attach <running container ID> /bin/bash

docker exec -it <running container ID> bash
Or
docker exec -it <running container ID> /bin/bash
```

Quit the container and keep it running

Press `ctrl + p` and `ctrl + q` simultaneously

```
ctrl <p> and ctrl <q>
```

Quit the container and stop it

```
exit
```

Stop the container

```
docker stop <container ID>
```

Restart the container

```
docker restart <container ID>
```

Delete the container (stop it first)

```
docker stop <container ID>  
docker rm <<container ID>
```

Remove multiple containers at once

```
docker rm + the beginning of the container ID  
docker rm ebc b e07c e3e68 6df8 3e55 bdae 73d1
```

Force and delete running containers

```
docker rm -f <container ID>
```

Delete an image

- Stop the container first
- Delete the container and delete the image
- i = image

```
docker stop <container ID>  
docker rm <container ID>  
docker rmi <container ID>
```

Force and delete and image

```
docker rmi <container ID> --force
```

Login in docker registry

```
docker login
```

Save changes

This is important if you want to save any modification or any change that you make inside the container like installing new packages, creating files, etc. This will create a new container with all your changes.

```
docker commit <container ID or name> <name-for-new-image>
```

Tag an image

Tag an image is the same as renaming the image. This is useful when creating a new image and you want to push it to docker hub or to make a copy of your image

```
docker tag <image name>:tag <username>/<repository name>:tag
OR
docker tag <image ID> <username>/<repository name>:tag

docker tag bf756fb1ae65 leonardtia/devops_repo:my-hello
docker tag my-centos:latest leonardtia/devops_repo:my-image-test
```

Save an image and tag is ready to be pushed to the docker hub

```
docker commit <container ID or name> <username >/<repository name>:tag
docker commit 793041fe6291 leonardtia/devops_repo:my-image-test
```

Push a new image to docker hub

```
docker push <username>/<image name>:tag
docker push leonardtia/devops-pro-repos:my_ubuntu
```

Delete an image in the docker hub repository?

- Click on the repository
- Click on see all
- Select the image, click on the action and delete

Copy file from container to docker host

This is to share data between the container and the host **In the container:**

```
docker run -it --name my-centos centos
docker ps
docker exec -it 5d9c798d0d44 bash
echo "This is a file from the container" >> /tmp/file1.txt
ctrl+p ctrl+q: tp exit the container
```

In the host machine:

```
docker cp my-centos:/tmp/file1.txt /file1.txt
docker <container ID>:/tmp/file1.txt /file1.txt

my-centos=container name
/tmp/file1.txt=path in the container
/file1.txt=location where we want to put our file on docker host
```

Copy from host to the containir

```
docker cp /file1.txt <container ID>:/tmp/file2.txt
docker cp /file1.txt my-centos:/tmp/file2.txt
docker exec -it 5d9c798d0d44 bash
```