

Introduction to Containers: Docker installation and deployment

1. Switch to root user permission

a. `sudo su`

2. Update the System

a. `apt update`

3. Install Docker container to the system

a. `apt install docker.io`

4. Find the location of the Docker binary

a. `which docker`

5. Print current version of docker

a. `docker --version`

6. Print all information about docker

a. `docker info`

7. Show current status of docker

a. `service docker status`

8. Pull image from Docker Hub

a. `docker pull image_name[:tag]`

b. `docker pull ubuntu`

9. Display all images in local system

a. `docker images`

10. Create and Start a Container:

a. `docker run [options] image_name`

b. `docker run -it --name my_container ubuntu`

11. List Running Containers:

a. `docker ps`

12. List All Containers (Including Exited):

a. `docker ps -a`

13. Stop a Running Container:

a. `docker stop container_id_or_name`

b. `docker stop my_container`

14. Start a Stopped Container:

a. `docker start container_id_or_name`

b. `docker start my_container`

15. Restart a Container:

a. `docker restart container_id_or_name`

b. `docker restart my_container`

16. Remove a Stopped Container:

a. `docker rm container_id_or_name`

b. `docker rm my_container`

17. Remove a Running Container:

a. To remove a running container, you need to stop it first and then remove it.

b. `docker stop container_id_or_name` `docker rm container_id_or_name`

c. `docker stop my_container` `docker rm my_container`

18. Inspect Container Details:

a. `docker inspect container_id_or_name`

b. `docker inspect my_container`

19. Attach your terminal's standard input and output to a running container

a. `docker attach my_container`

20. Install a webserver application such as 'apache' on a container, 'my_container'

a. `apt update`

b. `apt install apache2`

21. Create a new image from changes made to a running container

a. `docker commit [options] container_id|container_name new_image_name[:tag]`

b. `docker commit my_container my_custom_image:latest`

22. Remove an Image:

a. `docker rmi image_name[:tag]`

b. `docker rmi my_custom_image:latest`

23.Tag an Image:

- a. `docker tag source_image_name[:tag] target_image_name[:tag]`
- b. `docker tag my_custom_image:latest username/my_repository:latest`

24.Login to Docker Hub

- a. Docker login

25.Push an Image to Docker Hub:

- a. `docker push image_name[:tag]`
- b. `docker push username/my_repository:latest`