**Common Commands of Docker**

1. Retrieve and create a centOS container
   1. docker container run centos
   2. Notice that the container terminates immediately after creation
2. Check the running containers
   1. docker container ls
   2. Notice that no container is available
   3. docker container ls -a 🡪 This gives the list of all the containers
3. Run the container again and get bash of the running container
   1. docker container run -it centos:latest /bin/bash
   2. Notice the change in prompt
4. Perform some operations in the container
   1. ps -elf
   2. cat /etc/\*release\*
5. Exit the container without killing it by pressing together, Ctrl-PQ
6. Test if the container is still running
   1. docker container ls
7. Re-attach to the container,
   1. docker container exec -it [your container name] bash
8. Exit the container again like in step 5 and stop it.
   1. docker container stop [your container name]
9. Use the Inspect command to get information about the container
   1. docker container inspect [your container name/id]
10. We can narrow down the result of inspect by grep
    1. docker inspect [your container name] | grep IPAddress
11. Remove the container
    1. docker container rm [your container name]
12. check all the images in your system
    1. docker image ls
13. Remove images from host system
    1. docker image rmi [Image ID]
    2. Note that only those images which has no container in the host can be removed like this
14. download a fresh image.
    1. docker image pull ubuntu:17.10
15. create a container from the image just pulled
    1. docker container run ubuntu:17.10
16. run a container in attached mode
    1. docker container run ubuntu sleep 15
    2. Note that the control is not relinquished until the command runs
17. Run a container in detach mode
    1. docker container run -d ubuntu sleep 100
    2. we immediately get by the control and the container process runs in the background
18. re-attach the container
    1. docker container attach [container name/Id]
19. Start a container again
    1. docker container start [container ID]
20. Install Jenkins and map docker to host browser
    1. docker run -d Jenkins/Jenkins
21. Check log information
    1. docker logs [container ID]
22. Use inspect to find the IP of the container
23. Open browser and use the container IP to see Jenkins
    1. 172.17.0.2:8080
24. Stop Jenkins container
25. Start Jenkins with port mapping
    1. docker container run -p 8080:8080 jenkins/jenkins
26. Open the windows machine browser after a few seconds and type in the docker host IP
    1. Eg: 192.168.150.128:8080
    2. You can get host IP by running ifconfig on host terminal
27. Configure Jenkins by using the password provided in container logs
28. Create a dummy user, set password, create new item, login and logout a few times to see that the information is available
29. Stop the container and start it again to see that the information is available across container restarts
30. Stop the container and create a fresh container using step 25,26
31. Notice that the new container does not have the data saved previously
32. Create a local directory on host by mkdir command
    1. mkdir /home/netcom/Desktop/myDir
33. create a new container and map it with the directory just created
    1. docker container run -p 8080:8080 -v /home/netcom/Desktop/myDir:/var/jenkins\_home jenkins/jenkins
34. Repeat steps 26-29
35. Repeat step 33 to create a new container.
36. Notice that the data persists across containers this time