**What is Docker Container?**

Docker is a containerization platform that packages your application and all its dependencies together in the form of a docker container to ensure that your application works seamlessly in any environment.

Docker container runs applications created from the docker images.

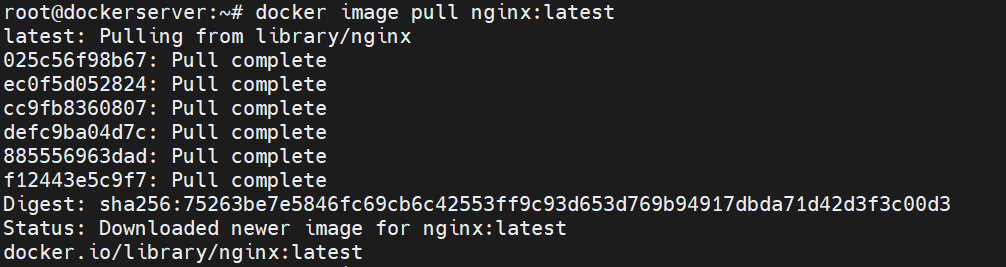
In order to create a container, we need a docker image. Without the docker image, the container can not be created.

docker image ls command will list all the images present in the host machine. As we had not pulled any image from the registry, it does not show any list.

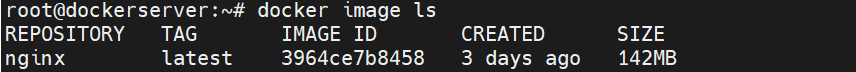


Now let us pull any image from the docker hub registry to create a container from it.

docker image pull image\_name command will pull image from docker hub registry.



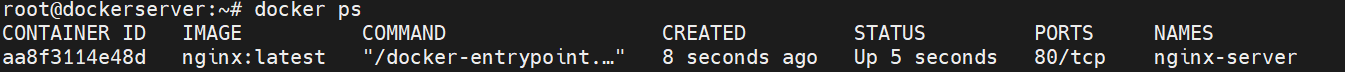
docker image ls command will list all the images present in the host machine. Above image has to be shown in the list.



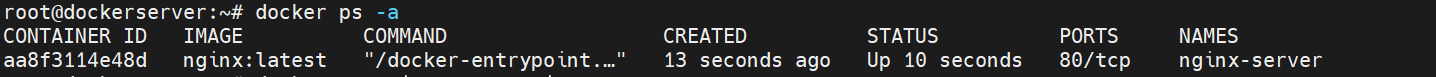
docker container run command will create the container, here -d option will execute the command in detached mode and –name will give the name of the container with the specified image.



docker ps command will list the containers in the host machine.



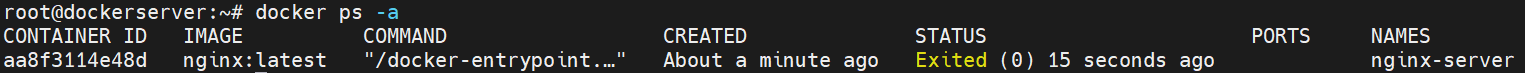
docker ps -a command will list all the containers in the host machine including stopped ones.



docker container stop container\_name command will stop the container from running.



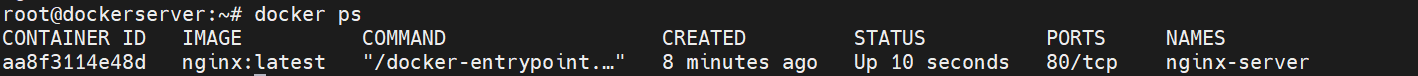
After stopping the container, the status of the container has to be exited with zero exit code, then the container has been successfully stopped.



docker container start container\_name command will start the container.



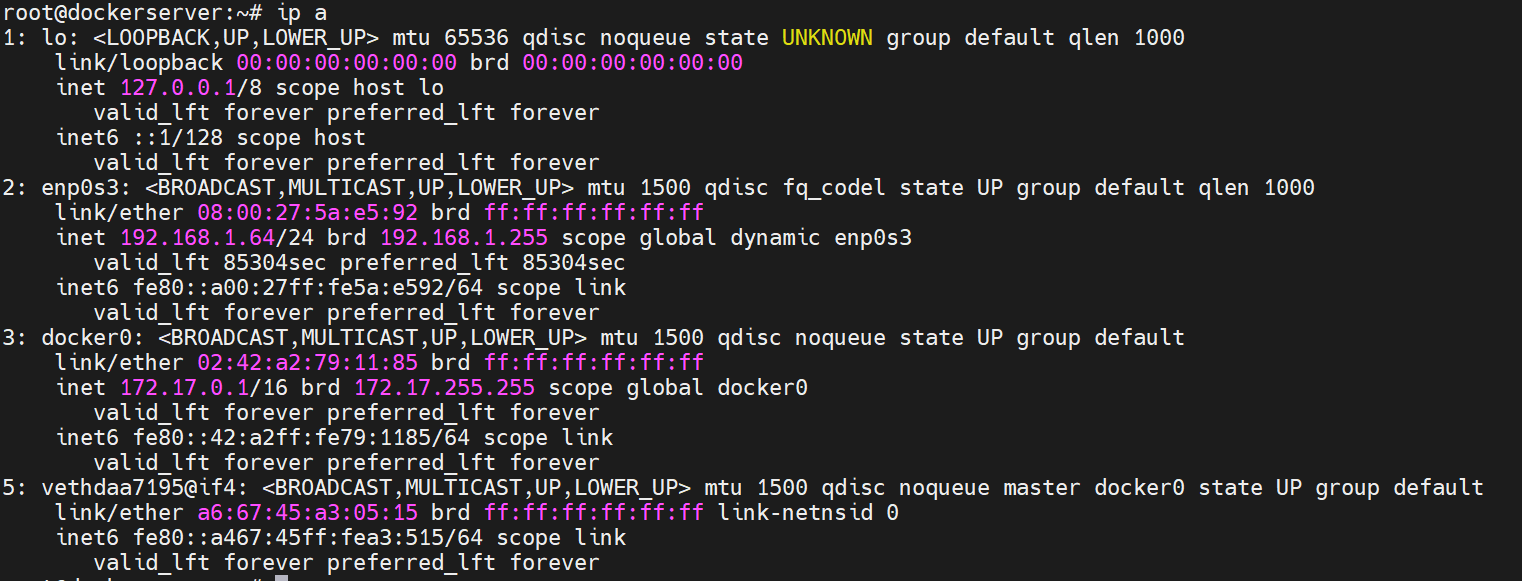
Now if we check docker container status, it is in running state.



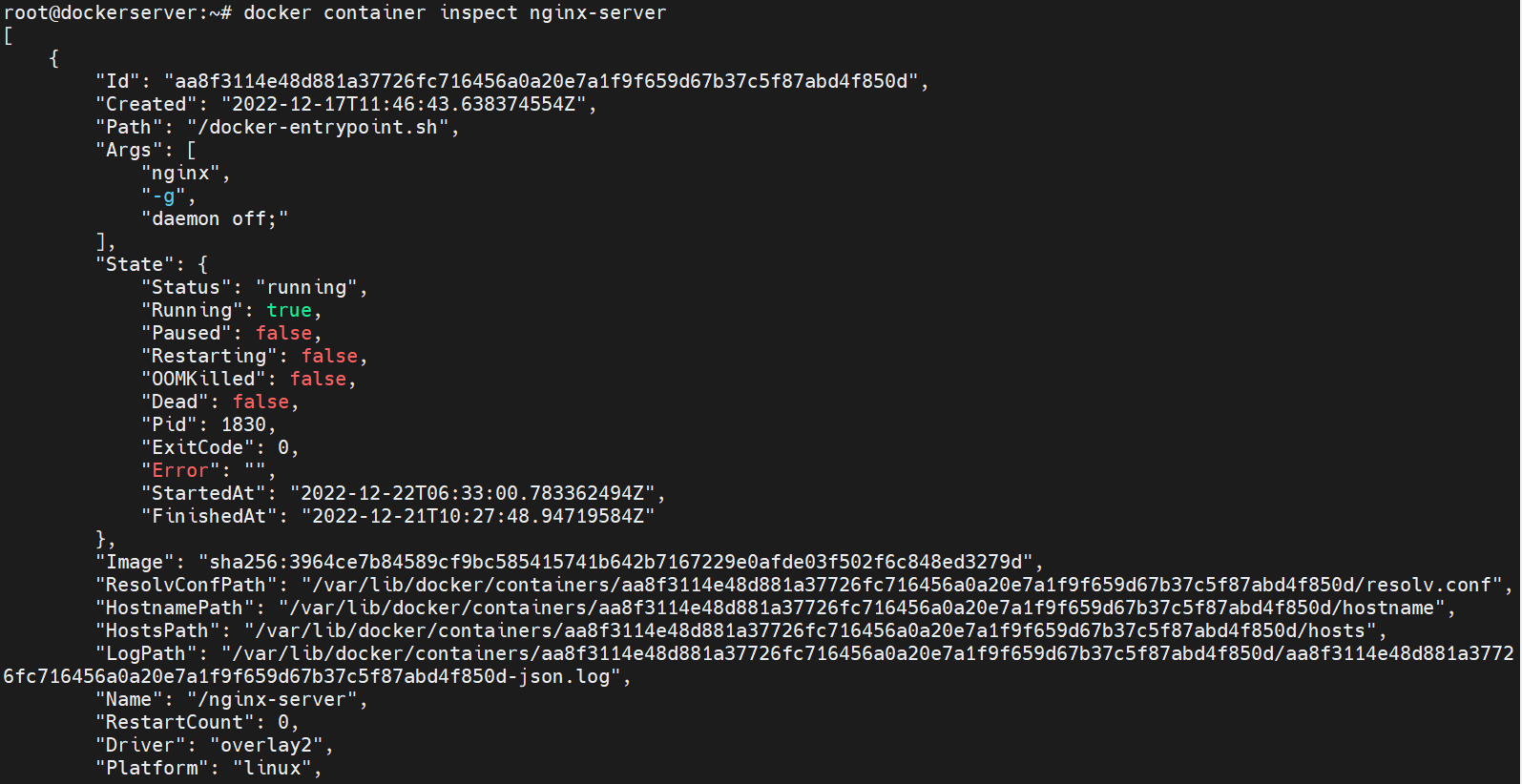
docker container restart container\_name command will restart the container.

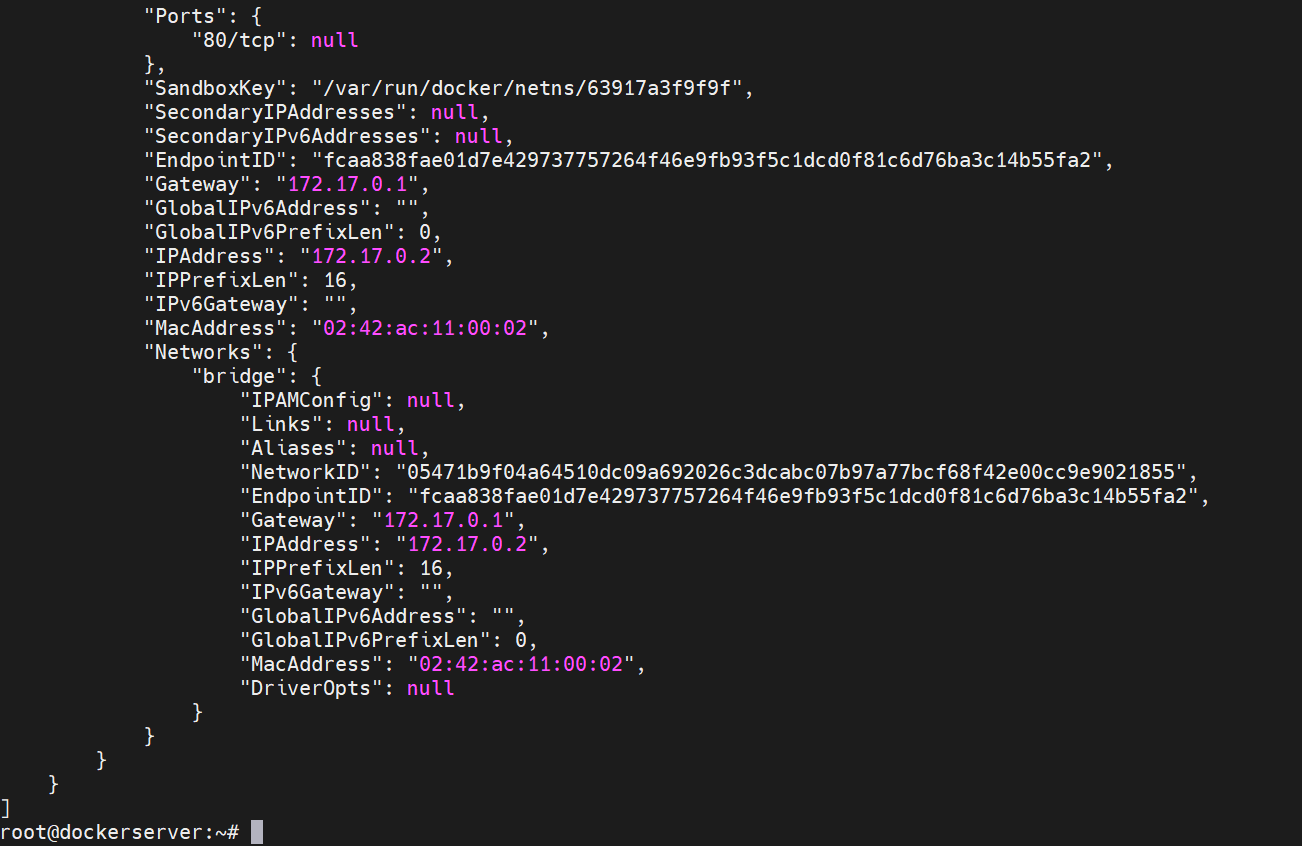


Containers will have a separate ip address starting with veth at the last as shown below rather than host machine ip and docker default ip address.



docker container inspect container\_name command will give the detailed information about the container.

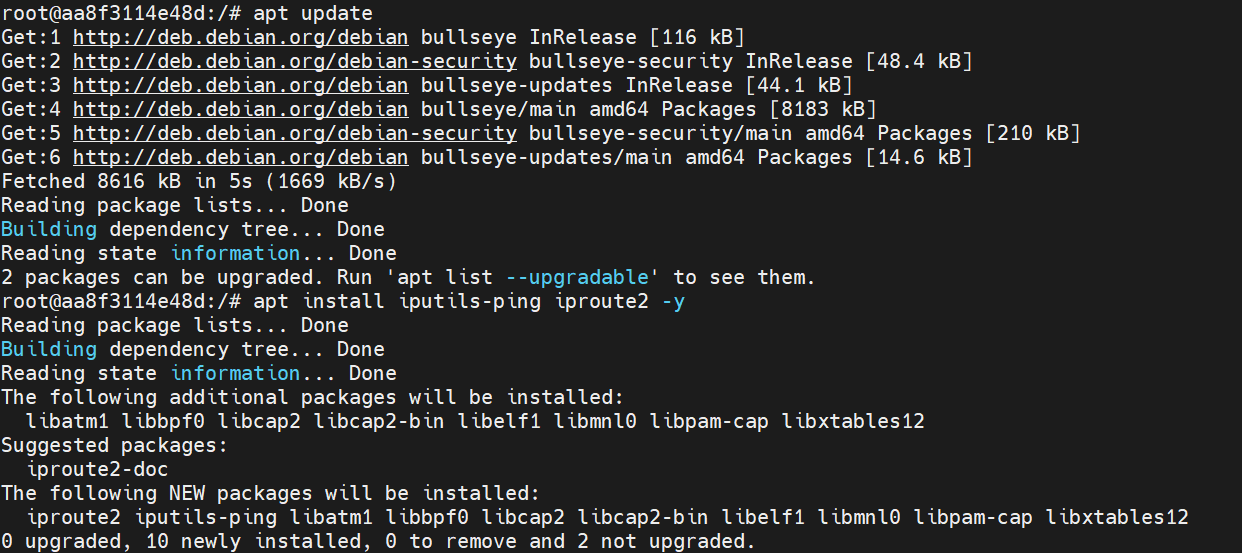




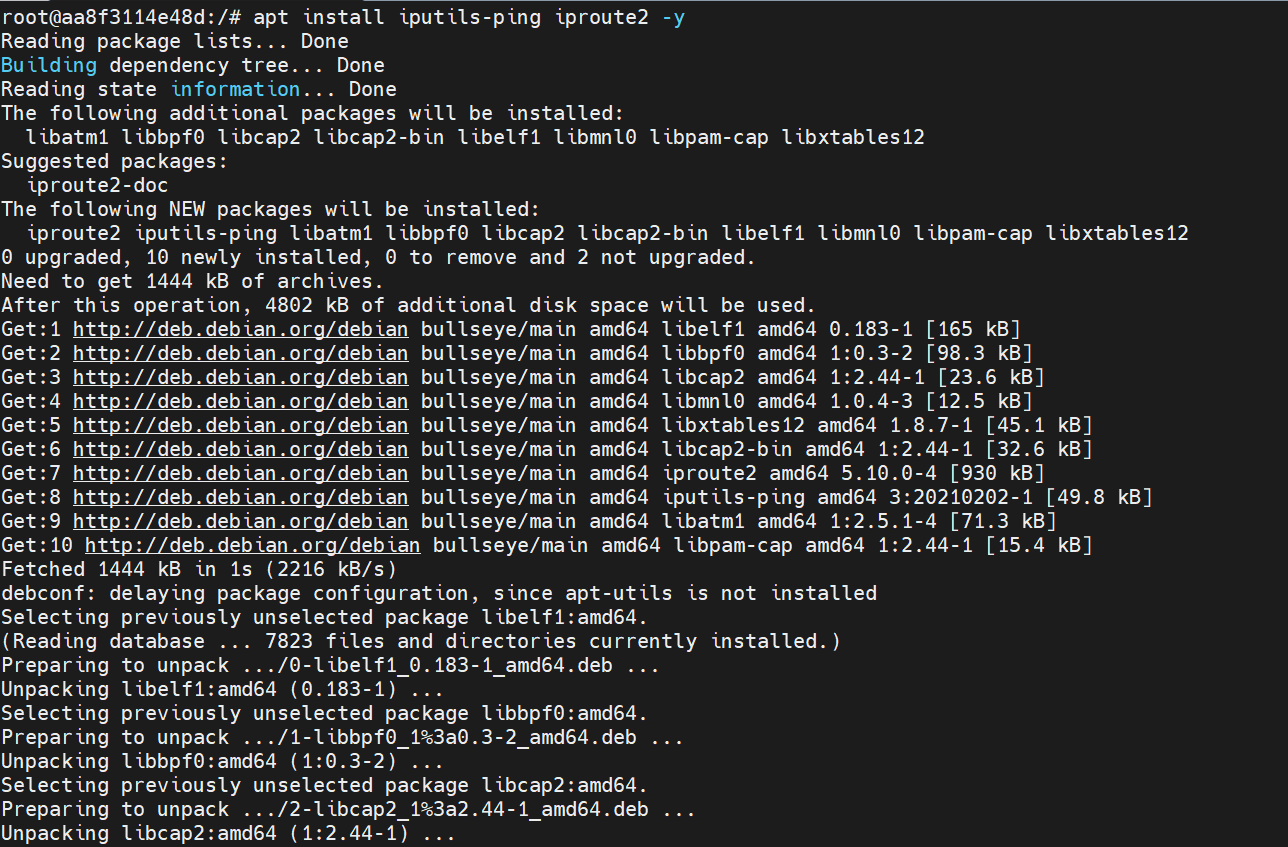
Docker exec -t -i container\_name bash command will login to the container using bash shell whereas options -t is for terminal mode and -i is for interactive mode.



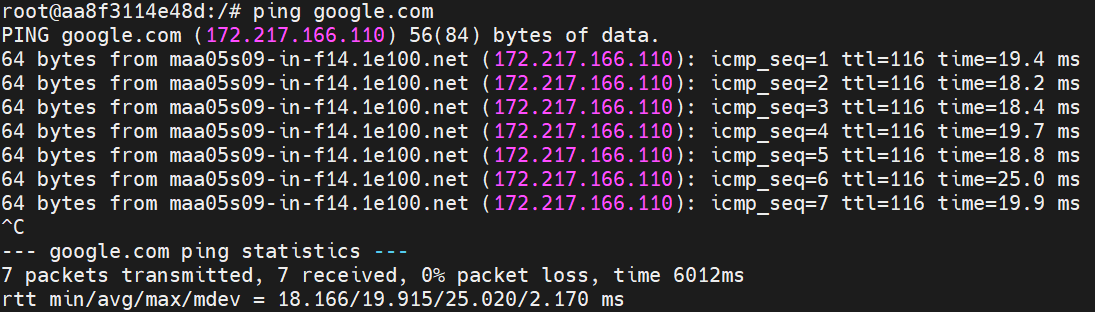
apt update command will update all the repositories. This shows that the container has network connection.



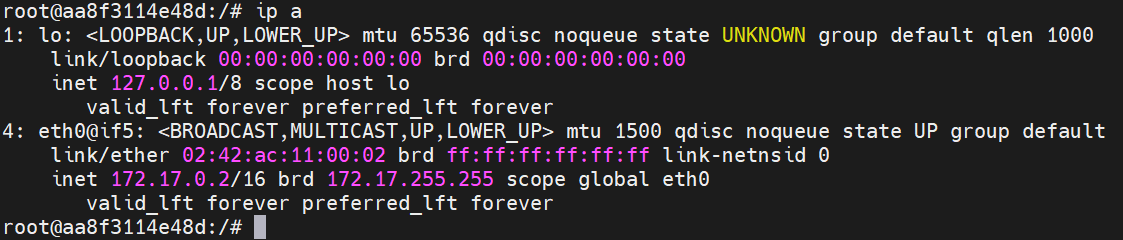
apt install iputils-ping iproute2 -y command will install network related commands like ping and ip.



Now let us check the internet connectivity using the ping command.



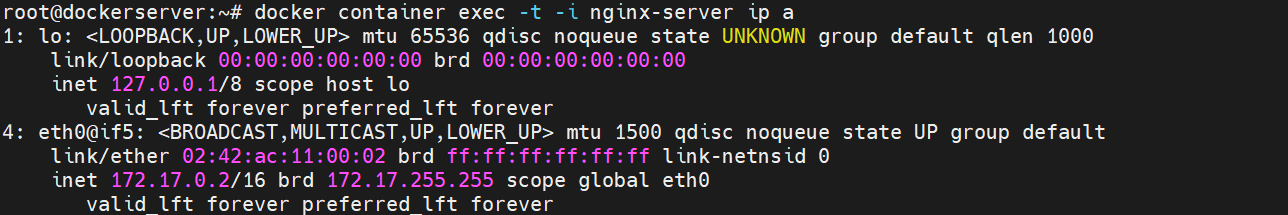
ip a command will give the container ip address.



exit command is used to exit from the container.



Instead of the bash command, we can pass the commands while login to the container itself. This will execute the command and give the output of the command soon after login to the respective container.



docker container stop container\_name command will stop the running container.



docker container rm container\_name command will remove the running container. If you have to delete the container, first the container has to be stopped.



docker ps command will give the list of running containers. As we have deleted the container, no containers are shown below.



docker container rm -f container\_name command will remove the running container, -f is for forceful removal of the running container.

Note: It is not at all the recommended approach to remove the running container forcefully.

