

COMMANDS	DESCRIPTION
yum install docker -y	to install docker
docker version	to check the docker version
systemctl status docker	to check the docker status
systemctl start docker	to start the docker
docker images	used to get list of images in our server
docker run image_name (ubuntu)	to get the image
docker ps -a	used to see all the containers
docker run -it --name cont_name image_name	used to create a container
docker attach cont_name/id	to enter into the container
docker stop cont_name/id	to stop the container
docker start cont_name/id	to start the docker
docker stop \$(docker ps -a -q)	to stop all containers
docker start \$(docker ps -a -q)	to start all containers
docker rm \$(docker ps -a -q)	to delete all containers
docker rm cont_name/id	to delete single container
docker rmi image_name/id	used to delete image
ctrl+p+q	used to exit from the container
docker run -it -d --name cont_name image_name	used to create a container (runs on background also)
docker run -it -d --name cont_name -p 8081:80 image_name	used to create a container with port number (8081 --> host port, 80 ----> container port)
docker exec -it cont_name "command"	used to perform the commands inside the container
docker container prune	used to remove unused containers
docker ps	used to see running containers only
docker ps -a -f "status=exited"	used to see only stopped containers
docker container ls	used to see running containers only
docker container ls -a	used to see all the containers
docker container ls -a -n 2	used to see latest 2 containers
docker container ls --latest	used to see the latest container
docker container ls -a -s	used to see all the containers along with the sizes
docker commit cont_name image_name	create image from the container
docker rename old_cont_name new_cont_name	used to rename the container
docker build -t image_name .	to build the docker file (. represents the path of the docker file)
docker build -t image_name:tag_name .	to build the docker file with tag
docker volume create volume_name	used to create a volume
docker volume rm volume_name	used to delete the volume
docker volume ls	used to see the list of volumes
docker volume inspect volume_name/id	used to inspect volume
docker volume prune	used to delete unused volumes
docker run -it --name cont_name -v /volume_name image_name	used to create a volume inside a container
docker run -it --name cont_name --privileged=true --volumes-from cont_name image_name	used to share the volumes to container
docker run -it --name cont_name --mount source=volume_name,destination=volume_name	used to mount the local volumes to container volumes
docker run -it --name cont_name -v local_volume:/cont_volume image_name	base volume shared to container
docker run -it --name cont_name -v \$("pwd"):/volume_name image_name	used directory as a volume inside the container
docker inspect cont_name/id	used to get the details of the container
docker image inspect image_name/id	used to get the detail of the image
docker network ls	used to get the list of networks
docker network create network_name	used to create a network
docker network connect network_name cont_name	used to attach a network to container
docker network disconnect network_name cont_name	used to disconnect the network from the container
docker network rm network_name	used to remove network
docker network prune	used to remove unused networks
docker run -it --name cont_name --network network_name image_name	used to create a container along with the network
docker login	used to login into dockerhub
docker tag image_name docker_hub_id/repo_name ex: (docker tag image-1 mustafa77:latest docker_hub_id/repo_name)	used to tag the image to docker hub (mustafa77 is docker hub id and repo-1 is repo_name)
docker push docker_hub_id/repo_name	used to push the image from local to docker hub
docker pull image_name	used to get the image from docker hub to local
docker swarm init --advertise-addr public_ip_of_manager	used to generate a token for adding worker nodes to swarm
docker swarm join-token manager	generate the manager token
docker swarm join-token worker	generates the worker token
docker node ls	used to see the list of nodes
docker service create --name service_name --replicas 2 --publish 8081:80 image_name	used to create a service
docker service ls	used to see list of services
docker service rm service_name	used to delete service
docker service update --image service_name	used to update the image of the service
docker service rollback service_name	used to rollback to prev service
docker service scale service_name=4	used to increase the replicas
docker service inspect service_name	to inspect a service
docker service logs service_name	used to get logs of a service
docker-compose up -d	used to execute the compose file

docker-compose down	remove the containers
docker-compose stop	used to stop the compose containers
docker-compose start	used to start the containers
docker-compose pause	used to pause the containers
docker-compose unpause	used to unpause the containers
docker-compose ps	used to get the compose containers
docker-compose build	used to build the images which are present in compose file
docker-compose images	used to get the images of the compose files
docker-compose logs	used to get the logs of compose file
docker-compose config	used to get the details of the compose file
docker-compose -f file_name.yml up -d	execute any compose file
docker stack deploy --config-name docker-compose.yml stack_name	used to deploy a stack file
docker stack ls	used to get list of stacks
docker stack ps stack_name	used to get the containers associated to our stack
docker stack services stack_name	used to get the services for our stack
docker stack rm stack_name	used to remove the stack
docker system df -v	used to see all the docker components information