Docker Cheat Sheet

by pratikshahp via cheatography.com/180680/cs/37579/

Docker Installation

apt update	Updates the package sources with the latest versions
apt install docker.io -y	Install docker in Ubutu Linux distribution
systemctl start docker	Stop docker services
systemctl stop docker	Stop docker services
systemctl restart docker	Stop and start docker services
systemctl status docker	Check the status of docker services

Container Operations

docker container create --name web -it alpine sh	Create a named container
docker container start web	Start a container
docker container stop web	Stop a container
docker container restart web	stop and start a container
docker container run -it --name myweb alpine sh	create and start a named container
Ctrl + p, Ctrl + q	Detach from the container
docker container attach myweb	Attach to a running container
docker container logs myweb	See the logs of the container
docker container rename myweb myalpine	Change the name of a container
docker container run -h alpine -it --rm alpine sh	Set the hostnameof a container
docker container run -it -w /var --rm alpine sh	Set the current working directory
docker container run -it --env "WEB_HO-ST=172.168.1.1" --rm alpine sh	Set the environment variables

Container Operations (cont)

docker container exec myalpine [command]	Create any process inside an already running container

Docker Management

docker version	prints the current version number
docker info	displays system wide information
docker login	Log in to a DockerHub registry
docker logout	Logout from DockerHub registry
docker inspect [object]	Display detailed information on one or more objects.
docker --help	list the help on any command
docker image [option]	To work with images
docker container [option]	To work with containers
docker volume [option]	To work with volumes
docker network [option]	To work with network

List

docker image ls	List of images
docker image ls --digests	List of image digests
docker container ls	List of running containers
docker container ls -a	List of all containers
docker container ls -q	Get only the containers IDs of containers
docker volume ls	List of volumes
docker network ls	List of networks

Docker Images

docker search nginx	List the available images on DockerHub
docker image pull nginx	Pull an image from DockerHub
docker image pull myregistry.com:-8080/alpine:latest	Pull an image from a private registry
vim Dokerfile FROM alpine RUN date > /data	**Dockerfile** to build an image
docker image build -t php/alpine:d-ockerfile .	Create an image from a Dockerfile
docker container run -it --name myimage alpine sh / # date > /data / # cat /data	Run a container and modify it.
docker container diff myimage	Check the difference with the base image
docker container commit myimage php/alpine:test	Save a running container as an image.
docker container run -it php/alpin-e:test sh cat /data	Run the created image as a container
docker image push php/alpine:test	Push the image to DockerHub registry

Docker Volumes

docker container run -itv /data --name myvol alpine sh	Create a directory on the host system and mount it under /data of the container
docker volume create --name myvolume	Create a named volume
docker container run -itv myvolume:/data --name cmyvol alpine sh	Mount a named volume
docker system df	Get the information about disk storage

Docker Networking

docker container run -d --name myweb -p 8080:80 nginx	Access the container from the external world
docker container run -d --name myweb1 -P nginx:alpine	Expose a container port to a random port of the host system
docker container run -it --network-=none alpine sh	To Request Docker daemon not to create an interface for the container
docker container run -it --network-=host alpine sh	share the network namespace of the host system with the container
docker container run -it --network-=container:myweb1 alpine sh	Share the network namespace of an existing container
docker network create mynetwork	Create a user defined network
docker container run -d --name nginxnet --network=mynetwork nginx:alpine	Create a container and attach it to the network
docker network connect mynetwork web	Attach a running container to a network
docker network disconnect mynetwork web	Disconnect from a specific network

Docker Resource Allocation

docker container run -it --ulimit nproc=10 --rm alpine sh	Set the maximum number of processes the container can create
docker container run -d --name cpulimit --cpuset-cpus="0" alpine top	Container can run only on CPU “0”.
docker container run -d --name memlimit --memory "200m" alpine top	Limit the maximum amount of memory usage

Docker Compose

curl -L "https://github.com/docker/compose/releases/download/1.27.4/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose	Install a docker compose
chmod +x /usr/local/bin/docker-compose	Set the executable
mkdir wordpress cd wordpress/ vim docker-compose.yaml	Create a folder and yaml file
docker-compose up	Start the application in the foreground
docker-compose up -d	Start the application in the background
docker-compose ps	List containers created by the application
docker-compose stop	Stop all the containers for the given application

Clean Up

docker image rm nginx:alpine	Remove an image locally
docker image prune	Clean an unused/dangling image
docker container rm -f web	Delete a running container
docker container rm myweb	Delete a stopped container
docker container prune	Remove all stopped containers
docker network rm mynetwork	Remove a specific network
docker network prune	Remove all networks which are not referenced by any containers
docker volume rm myvolume	Remove a specific volume
docker volume prune	Remove all unused volume

Clean Up (cont)

docker-compose rm	Remove the containers for the given application
docker-compose down -v	Delete the application completely with its containers, networks, volumes and images
docker system prune	Remove unused images, stopped containers, and unused networks