1 Docker CheatSheet   KUBERNETES

- PDF Link: cheatsheet-docker-A4.pdf, Category: tools

- Blog URL: `https://cheatsheet.dennyzhang.com/cheatsheet-docker-A4`

- Related posts: Kubernetes Yaml, #denny-cheatsheets

File me Issues or star this repo.

1.1 Docker start service

Name	Summary
Start a ubuntu test env	`docker run ubuntu:16.04 /bin/echo hello world`
Start a centos test env	`docker run centos:centos6 /bin/echo hello world`
Start a nexus server	`docker run -d -p 8082:8081 --name nexus -v /data/nexus-data:/nexus-data sonatype/docke`
Start a nginx server	`docker run -t -d -p 8080:80 --name nginx-test nginx`
Start a sshd server	`docker run -t -d --privileged -p 5022:22 denny/sshd:latest /usr/sbin/sshd -D`
Start a jenkins server	`docker run -p 8080:8080 denny/jenkins:1.0`
Start a ftp server	`docker run -t -d -p 21:21 -p 20:20 -e USERNAME=${username} -e PASSWORD=${password} den`

1.2 Docker Containers

Name	Summary
Start docker container	`docker run -p 4000:80 imgname`
Start docker container in detached mode	`docker run -d -p 4000:80 imgname`
Enter a running container	`docker exec -it [container-id] sh`
Stop container	`docker stop <hash>`
Remove container	`docker rm <hash>`
Remove all containers	`docker rm $(docker ps -a -q)`
Force shutdown of one given container	`docker kill <hash>`
Login to docker hub	`docker login`
Run image from a given tag	`docker run username/repo:tag`
Tail container logs	`docker logs --tail 5 $container_name`
Check container healthcheck status	`docker inspect --format '{{.State.Health}}' $container_name`
List all containers	`docker ps`, `docker ps -a`
List containers by labels	`docker ps --filter "label=org.label-schema.group"`

1.3 Docker Images

Name	Summary
List all images	`docker images`, `docker images -a`
Create docker image	`docker build -t denny/image:test .`
Docker push a tagged image to repo	`docker push username/repo:tag`
Show the history of an image	`docker history <image_name>`
Export image to file	`docker save <image_name> > my_img.tar`
Load image to local registry	`docker load -i my_img.tar`
Tag <image>	`docker tag <image> username/repo:tag`

1.4 Docker Cleanup

Name	Summary
Remove the specified image	`docker rmi <imagename>`
Remove all docker images	`docker rmi $(docker images -q)`
Remove orphaned docker volumes	`docker volume rm $(docker volume ls -qf dangling=true)`
Remove dead containers	`docker rm $(docker ps --filter status=dead -qa)`
Remove exited containers	`docker rm $(docker ps --filter status=exited -qa)`
Delete all containers	delete-all-containers.sh
Remove unused docker images	delete-unused-images.sh

1.5   Ubuntu docker: Install missing packages

Name	Summary
Pull ubuntu docker image	`docker pull ubuntu`
man: command not found	`apt-get update`, `apt-get install man`
ping: command not found	`apt-get update`, `apt-get install iputils-ping`
dig: command not found	`apt-get install dnsutils`

1.6   Docker Socket file

Name	Summary
Links	Link: Develop with Docker Engine SDKs and API
Run container mounting socket file	`docker run -v /var/run/docker.sock:/var/run/docker.sock -it alpine sh`
A different docker socket file	`export DOCKER_HOST=unix:///my/docker.sock`
List containers	`curl -XGET --unix-socket /var/run/docker.sock http://localhost/containers/js`
Stop container	`curl -XPOST --unix-socket /var/run/docker.sock http://localhost/containers/<`
Start container	`curl -XPOST --unix-socket /var/run/docker.sock http://localhost/containers/<`
List events	`curl --unix-socket /var/run/docker.sock http://localhost/events`
Create container	`curl -XPOST --unix-socket /var/run/docker.sock -d '{"Image":"nginx:alpine"}'`

1.7   Docker Compose

Name	Summary
Change entrypoint to run nothing	`entrypoint:  ["tail", "-f", "/dev/null"]`
Change restart policy	`restart:  always`, Link: Compose file version 3 reference
Mount file as volume	`$PWD/httpd/httpd.conf:/usr/local/apache2/conf/httpd.conf:ro` sample-mount-file.ym
Start compose env	`docker-compose up`, `docker-compose up -d`
Stop compose env	`docker-compose down`, `docker-compose down -v`
Check logs	`docker-compose logs`

1.8   Container Runtime

Name	Summary
dockerd	
rkt	a pod-native container engine for Linux.
CRI-O	Open Container Initiative-based implementation of Kubernetes Container Runtime Interface

1.9   Resource Reference

Name	Summary
Docker SDK	`https://docs.docker.com/develop/sdk/examples/`
Docker REST API	`https://docs.docker.com/engine/api/v1.27/#tag/Container`
Docker Hub auto build	`https://docs.docker.com/docker-hub/builds/#build-statuses-explained`
Docker files	`/var/lib/docker`, `/var/lib/docker/devicemapper/mnt`

1.10   More Resources

License: Code is licensed under MIT License.