1 Docker CheatSheet KUBERNETES

- PDF Link: cheatsheet-docker-A4.pdf
- Blog URL: `https://cheatsheet.dennyzhang.com/cheatsheet-docker-A4`
- Category: tools

File me Issues or star this repo.
See more CheatSheets from Denny: #denny-cheatsheets

1.1 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` samples/mount-file.y
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`

1.2 Check Status

Name	Summary
Tail container logs	`docker logs --tail 5 $container_name`
Check container healthcheck status	`docker inspect --format '{{.State.Health}}' $container_name`
List containers	`docker ps`
List all containers	`docker ps -a`
List containers by labels	`docker ps --filter "label=org.label-schema.group"`
List all images	`docker images -a`

1.3 Container Basic

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`
Tag <image>	`docker tag <image> username/repo:tag`
Docker push a tagged image to repo	`docker push username/repo:tag`
Run image from a given tag	`docker run username/repo:tag`
Create docker image	`docker build -t denny/image:test .`

1.4 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 -ti alpine sh`
A different docker socket file	`export DOCKER_HOST=unix:///my/docker.sock`
List containers	`curl -XGET --unix-socket /myrun/docker.sock http://localhost/containers/json`
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"}'`

1.5 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)`

1.6 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`

1.7 Scripts

- Docker build

```
cd "$dir" && cat "$dockerfile" | docker build -t "$docker_image" -f - .
```

- Use google docker registry

```
docker build -t gcr.io/<MY_REPO>/<MY_IMAGE>:<MY_TAG> .
gcloud docker -- push gcr.io/<MY_REPO>/<MY_IMAGE>:<MY_TAG>
```

- Delete all containers

delete-all-containers.sh

```
curl -L https://raw.githubusercontent.com/dennyzhang/cheatsheet-docker-A4/master/delete-all-containers.sh | b
```

- Install test kit

container-install-devkit.sh

```
apt-get -y update
apt-get install -y curl netcat

curl -L https://raw.githubusercontent.com/dennyzhang/cheatsheet-docker-A4/master/container-install-devkit.sh
```

- Clean up Disk

Remove All Useless Resources.

```
docker ps --filter status=exited -aq | xargs -r docker rm -v
```

Remove unused docker images

```
docker rmi $(docker images | grep "<none>"  | awk -F' ' '{print $3}')
```

Remove intermediate containers generated during docker build

```
docker ps -a | grep "/bin/sh -c" |  awk -F' ' '{print $1}' | xargs docker rm
```

Remove Image with <none> string

```
echo "Remove docker images with <none> string"
if docker images | grep none | tee; then
   docker rmi $(docker images | grep "<none>"  | awk -F' ' '{print $3}') | tee
fi
```

1.8 More Resources