

Docker Cheat Sheet



dockerlux.github.io

@gcuisinier

General Usage

Start a container in background

```
$> docker run -d jenkins
```

Start an interactive container

```
$> docker run -it ubuntu bash
```

Start a container automatically removed on stop

```
$> docker run --rm ubuntu bash
```

Export port from a container

```
$> docker run -p 80:80 -d nginx
```

Start a named container

```
$> docker run --name mydb redis
```

Restart a stopped container

```
$> docker start mydb
```

Stop a container

```
$> docker stop mydb
```

Add metadata to container

```
$> docker run -d \
label=traefik.backend=jenkins jenkins
```

Build Images

Build an image from Dockerfile in current directory

```
$> docker build --tag myimage .
```

Force rebuild of Docker image

```
$> docker build --no-cache .
```

Convert a container to image

```
$> docker commit c7337 myimage
```

Remove all unused images

```
$> docker rmi $(docker images \
-q -f "dangling=true")
```

Debug

Run another process in running container

```
$> docker exec -it c7337 bash
```

Show live logs of running daemon container

```
$> docker logs -f c7337
```

Show exposed ports of a container

```
$> docker port c7337
```

Volumes

Create a local volume

```
$> docker volume create --name myvol
```

Mounting a volume on container start

```
$> docker run -v myvol:/data redis
```

Destroy a volume

```
$> docker volume rm myvol
```

List volumes

```
$> docker volume ls
```

Create a local network

```
$> docker network create mynet
```

Attach a container to a network on start

```
$> docker run -d --net mynet redis
```

Connect a running container from a network

```
$> docker network connect mynet c7337
```

Disconnect container to a network

```
$> docker network disconnect mynet c7337
```

Manage Containers

List running containers

```
$> docker ps
```

List all containers (running & stopped)

```
$> docker ps -a
```

Inspect containers metadatas

```
$> docker inspect c7337
```

List local available images

```
$> docker images
```

Delete all stopped containers

```
$> docker rm $(docker ps --filter status=exited -q)
```

List all containers with a specific label

```
$> docker ps --filter label=traefik.backend
```

Query a specific metadata of a running container

```
$> docker inspect -f '{{ .NetworkSettings.IPAddress }}' c7337
```

Legend

Image name

```
redis, jenkins, nginx
```

Container name or commit ID

```
mydb #name
```

```
c7337 #commit id
```