<https://docs.docker.com/get-started/#recap-and-cheat-sheet>

Recap and cheat sheet

## List Docker CLI commands

docker

docker container --help

## Display Docker version and info

docker --version

docker version

docker info

## Execute Docker image

docker run hello-world

## List Docker images

docker image ls

## List Docker containers (running, all, all in quiet mode)

docker container ls

docker container ls --all

docker container ls -aq

## Conclusion of part one

Containerization makes [CI/CD](https://www.docker.com/use-cases/cicd) seamless. For example:

* applications have no system dependencies
* updates can be pushed to any part of a distributed application
* resource density can be optimized.

With Docker, scaling your application is a matter of spinning up new executables, not running heavy VM hosts.

<https://docs.docker.com/get-started/part2/#dockerfile>

Git : <https://stackoverflow.com/questions/11696295/rejected-master-master-non-fast-forward>