



Docker for Beginners

Four concepts. Ten commands. Self-hosted services for life.

4

CONCEPTS

10

COMMANDS

~5min

FIRST CONTAINER

Linux

SHARED KERNEL

• The 4 concepts you need

TEMPLATE

Image

The blueprint. postgres:16.
Read-only.

Container

Running instance. Read-write
while running. Many
containers from one image.

Volume

Persistent storage that
survives container restart.
Where your data lives.

Network

How containers talk. Same
Compose file = name-
resolution by service.

• The 10 commands (95% of usage)

docker run -p 8080:80 image

start a new container

docker ps · docker ps -a

list running / all

docker stop name · rm name

stop / remove

docker logs -f name

tail logs live

docker exec -it name bash

shell into container

docker images · pull image

list / fetch

docker compose up -d

start whole stack, detached

docker compose down

stop + remove stack

PRO TIP

Use Compose for anything > 1
container. Even single-container
deploys benefit from the
documentation a Compose file
provides.

COMMON MISTAKES

Bundling Postgres + app in one
container. Pinning to :latest
(changes underneath you). Storing
real data inside a container with no
volume.

MENTAL MODEL CHECKLIST

1 One service per container

Easier to update / debug

2 Pin image versions

postgres:16, not :latest

3 Volumes for state

DB data, user uploads, configs

4 restart: unless-stopped

Comes back after reboot

5 Treat containers as cattle

Recreate freely

FIRST CONTAINERIZED SERVICE

1 brew install --cask docker

macOS — Docker Desktop

2 mkdir vaultwarden && cd

One folder per service

3 Write docker-compose.yml

image + env + volumes + ports

4 docker compose up -d

Done. Visit localhost:port.

Full guide: djenterprises.ai/blog/docker-for-beginners

DJENTERPRISES · AI CONSULTING & IOS APP STUDIO