



Docker CHEAT SHEET



Building Images

```
$ docker build -t name:tag .
```

Build from Dockerfile

```
$ docker build --no-cache -t name:tag .
```

Fresh build (no cache)

```
$ docker build --build-arg KEY=VAL -t name:tag .
```

Pass build args

```
$ docker tag source new_tag
```

Tag before push

```
$ docker history name:tag
```

Show layers & sizes



Working with Images

```
$ docker images
```

List local images

```
$ docker pull / push name:tag
```

Pull / Push to registry

```
$ docker rmi name:tag
```

Remove image

```
$ docker image prune -a
```

Remove ALL unused images

```
$ docker save -o file.tar name:tag
```

Export image to file

```
$ docker load -i file.tar
```

Import image from file



Container Lifecycle

```
$ docker run -d --name n -p 80:80 img
```

Detached + port mapping

```
$ docker run -it img bash
```

Interactive + shell

```
$ docker start | stop | restart name
```

Start / Stop / Restart

```
$ docker rm name
```

Remove stopped container

```
$ docker rm -f name
```

Force-remove running

```
$ docker container prune
```

Remove all stopped

```
$ docker stop $(docker ps -q)
```

Stop all running

👁️ Diagnostics & Debugging

```
$ docker ps -a
```

Running containers (all)

```
$ docker logs -f --tail 50 name
```

Last 50 lines + follow

```
$ docker logs --since 30m name
```

Logs from last 30 min

```
$ docker exec -it name bash
```

Enter container (bash / Alpine: sh)

```
$ docker inspect name
```

Full JSON config

```
$ docker stats
```

Live CPU/RAM usage

```
$ docker top name
```

Processes inside

```
$ docker port name
```

Mapped ports



System Cleanup

```
$ docker system prune
```

Remove unused containers/networks/images

```
$ docker system prune -a --volumes
```

HARD RESET (incl. volumes)

```
$ docker builder prune
```

Clear build cache

```
$ docker system df
```

Disk usage breakdown



Docker Compose (v2)

```
$ docker compose up -d
```

Start in background

```
$ docker compose up -d --build
```

Rebuild + start

```
$ docker compose down
```

Stop + remove containers/networks

```
$ docker compose down -v
```

ALSO remove volumes (data loss!)

```
$ docker compose logs -f
```

Live logs (all services)

```
$ docker compose ps
```

Service status

```
$ docker compose exec service bash
```

Enter service

```
$ docker compose restart service
```

Quick restart

```
$ docker compose pull [service]
```

Pull latest images, e.g. `docker compose pull db`

```
$ docker compose build
```

Build images defined in compose

```
$ docker compose run service cmd
```

One-off command in service (e.g. migrations)

```
$ docker compose config
```

Validate & view resolved `compose.yml`



Networks & Volumes

```
$ docker network ls
```

List networks

```
$ docker network inspect bridge
```

Network details (IPs)

```
$ docker network create name
```

Create network

```
$ docker volume ls
```

List volumes

```
$ docker volume rm name
```

Remove volume

```
$ docker volume prune
```

Remove unused volumes

```
$ docker cp cont:/path host_path
```

Container → Host

```
$ docker cp host_path cont:/path
```

Host → Container

💡 Pro Tips (~/.bashrc or ~/.zshrc)

```
$ alias dps='docker ps'
```

```
$ alias dcu='docker compose up -d'
```

```
$ alias dcd='docker compose down'
```

```
$ alias dlog='docker logs -f --tail 50'
```

```
$ alias dexec='docker exec -it'
```

```
$ alias dstop='docker stop $(docker ps -q)'
```

```
$ alias dclean='docker system prune -a --volumes'
```

Summary

- 🏗️ Building Images
- 📦 Working with Images
- ▶️ Container Lifecycle
- 👁️ Diagnostics & Debugging
- 🧹 System Cleanup
- 🐙 Docker Compose (v2)
- 🌐 Networks & Volumes
- 💡 Pro Tips (~/.bashrc or ~/.zshrc)



Found this useful?

Save this Docker CHEAT SHEET for later

Follow for more developer content

Like • Share • Repost