

■ Docker Command Cheat Sheet

> A clean and complete cheat sheet for working with Docker, covering everything from basic commands to advanced troubleshooting and networking.

■ Images

Search for images

```
docker search python
docker search --filter is-official=true python
```

Pull an image

```
docker pull python
docker pull python:3.12-rc-bookworm
```

List images

```
docker images
```

Remove an image

```
docker rmi image_name
```

Show image history

```
docker history image_name
```

■■ Build

Build from Dockerfile in current directory

```
docker build .
```

Specify a Dockerfile

```
docker build -f Dockerfile.custom .
```

Disable cache

```
docker build --no-cache .
```

Remove intermediate containers

```
docker build --rm=true .
```

Force remove intermediate containers even if build fails

```
docker build --force-rm=true .
```

■ Containers

Run a container

```
docker run -it ubuntu /bin/bash
```

Run with port mapping

```
docker run -d -p 8080:80 nginx
```

Run with volume

```
docker run -v /host:/container ubuntu
```

List all containers

```
docker ps -a
```

Start/Stop/Restart container

```
docker start container_name  
docker stop container_name  
docker restart container_name
```

Remove a container

```
docker rm container_name
```

Remove all stopped containers

```
docker container prune
```

■■ Inspect / Logs / Stats

Inspect detailed info (JSON)

```
docker inspect container_name
```

Show logs from container

```
docker logs container_name
```

Show logs with timestamps and tail

```
docker logs --timestamps --tail 50 container_name
```

Follow logs live

```
docker logs -f container_name
```

Show resource usage

```
docker stats container_name
```

View process tree inside container

```
docker top container_name
```

■ Networking

List Docker networks

```
docker network ls
```

Inspect a network

```
docker network inspect bridge
```

Create a new network

```
docker network create my-net
```

Run container on a custom network

```
docker run --network=my-net alpine ping 8.8.8.8
```

Connect/Disconnect container to/from network

```
docker network connect my-net container_name  
docker network disconnect my-net container_name
```

Get container IP address

```
docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}'  
container_name
```

Get container MAC address

```
docker inspect -f '{{range .NetworkSettings.Networks}}{{.MacAddress}}{{end}}'  
container_name
```

■ Volumes

List volumes

```
docker volume ls
```

Create a volume

```
docker volume create my-vol
```

Inspect a volume

```
docker volume inspect my-vol
```

Remove a volume

```
docker volume rm my-vol
```

Prune all unused volumes

```
docker volume prune
```

■ Dockerfile Essentials

Base image

```
FROM ubuntu:22.04
```

Set working directory

```
WORKDIR /app
```

Copy files

```
COPY . .
```

Run command during build

```
RUN apt-get update && apt-get install -y curl
```

Set environment variable

```
ENV APP_ENV=production
```

Default command

```
CMD ["python3", "app.py"]
```

■ Clean Up

Remove all stopped containers

```
docker container prune
```

Remove all unused images

```
docker image prune
```

Remove unused volumes and networks

```
docker system prune --volumes
```

■ Security Tips

Scan an image for vulnerabilities (requires Docker Scan plugin)

```
docker scan image_name
```

Run with limited capabilities

```
docker run --cap-drop=ALL --cap-add=NET_BIND_SERVICE alpine
```

Read-only filesystem

```
docker run --read-only nginx
```

■ Miscellaneous

Docker help

```
docker --help  
docker run --help
```

Get Docker version

```
docker version
```

Get system-wide info

```
docker info
```

Execute a command inside a running container

```
docker exec -it container_name /bin/bash
```