Docker Cheat Sheet

Installation

Install Docker Engine (https://docs.docker.com/engine/install/, Linux only) or Docker Desktop (https://docs.docker.com/desktop/, Linux, macOS and Windows).

Container commands

Command	Description
`docker run <image>`	Create and run a new container
`docker run -p 8080:80 <image>`	Publish container port 80 to host port 8080
`docker run -d <image>`	Run a container in the background
`docker run -v <host>:<container> <image>`	Mount a host directory to a container
`docker ps`	List currently running containers
`docker ps --all`	List all containers (running or stopped)
`docker logs <container_name>`	Fetch the logs of a container
`docker logs -f <container_name>`	Fetch and follow the logs of a container
`docker stop <container_name>`	Stop a running container
`docker start <container_name>`	Start a stopped container
`docker rm <container_name>`	Remove a container

Executing commands in a container

Command	Description
`docker exec <container_name> <command>`	Execute a command in a running container
`docker exec -it <container_name> bash`	Open a shell in a running container

Image commands

Command	Description
`docker build -t <image> .`	Build a new image from the Dockerfile in the current directory and tag it
`docker images`	List local images
`docker rmi <image>`	Remove an image

Container registry commands

Command	Description
`docker login`	Login to Docker Hub
`docker login <server>`	Login to another container registry
`docker logout`	Logout of Docker Hub
`docker logout <server>`	Logout of another container registry
`docker push <image>`	Upload an image to a registry
`docker pull <image>`	Download an image from a registry
`docker search <image>`	Search Docker Hub for images

System commands

Command	Description
`docker system df`	Show Docker disk usage
`docker system prune`	Remove unused data
`docker system prune -a`	Remove all unused data

Docker Compose

Command	Description
`docker compose up`	Create and start containers
`docker compose up -d`	Create and start containers in background
`docker compose up --build`	Rebuild images before starting containers
`docker compose stop`	Stop services
`docker compose down`	Stop and remove containers and networks
`docker compose ps`	List running containers
`docker compose logs`	View the logs of all containers
`docker compose logs <service>`	View the logs of a specific service
`docker compose logs -f`	View and follow the logs
`docker compose pull`	Pull the latest images
`docker compose build`	Build or rebuild services
`docker compose build --pull`	Pull latest images before building

Dockerfile instructions

Instruction	Description
`FROM <image>`	Set the base image
`FROM <image> AS <name>`	Set the base image and name the build stage
`RUN <command>`	Execute a command as part of the build process
`RUN ["exec", "param1", "param2"]`	Execute a command as part of the build process
`CMD ["exec", "param1", "param2"]`	Execute a command when the container starts
`ENTRYPOINT ["exec", "param1"]`	Configure the container to run as an executable
`ENV <key>=<value>`	Set an environment variable
`EXPOSE <port>`	Expose a port
`COPY <src> <dest>`	Copy files from source to destination
`COPY --from=<name> <src> <dest>`	Copy files from a build stage to destination
`WORKDIR <path>`	Set the working directory
`VOLUME <path>`	Create a mount point
`USER <user>`	Set the user
`ARG <name>`	Define a build argument
`ARG <name>=<default>`	Define a build argument with a default value
`LABEL <key>=<value>`	Set a metadata label
`HEALTHCHECK <command>`	Set a healthcheck command

See https://docs.docker.com/engine/reference/builder/ for the full Dockerfile reference.

Example compose.yaml

Note: this file used to be called `docker-compose.yaml`, but now `compose.yaml` is preferred.

```yaml
services:
  service1:
    image: <image>
    build: .
    volumes:
      - .:/code:ro
    ports:
      - "8000:80"
    environment:
      KEY: value
```

Docker Compose file reference

Key	Description
`name`	Set the name of the project
`services`	A list of services defined in the file
`services.<name>.image`	Set the image to use or build
`services.<name>.build`	Build context and options
`services.<name>.build.context`	Build context (default is the current directory)
`services.<name>.build.dockerfile`	Dockerfile to use (default is `Dockerfile`)
`services.<name>.build.target`	Build stage to use
`services.<name>.build.args`	Build arguments
`services.<name>.command`	Override the default command for the container
`services.<name>.entrypoint`	Override the default entrypoint for the container
`services.<name>.volumes`	Mount volumes in the container
`services.<name>.ports`	Publish container ports to the host
`services.<name>.environment`	Set environment variables in the container
`services.<name>.restart`	Restart policy (no/always/on-failure/unless-stopped)
`services.<name>.scale`	Set the number of containers to run
`services.<name>.networks`	List of networks to connect the container to
`services.<name>.depends_on`	List of services to start before this service
`services.<name>.labels`	Set metadata labels for the container
`networks`	A list of networks defined in the file
`networks.<name>.driver`	Set the network driver
`networks.<name>.external`	Don't create the network, use an existing one
`volumes`	A list of volumes defined in the file
`volumes.<name>.name`	Set the name of the volume
`volumes.<name>.driver`	Set the volume driver
`configs`	A list of configs defined in the file
`secrets`	A list of secrets defined in the file

See https://docs.docker.com/compose/compose-file/ for the full Docker Compose file reference.