

Basic Docker Commands

<code>docker search</code> <i>searchterm</i>	Search Docker Hub for images
<code>docker pull</code> <i>user/image</i>	Downloads an image from Docker Hub
<code>docker login</code>	Authenticate to Docker Hub (or other Docker registry)
<code>docker push</code> <i>user/image</i>	Uploads an image to Docker Hub. <i>You must be authenticated to run this command</i>
<code>docker info</code>	Retrieves docker configuration
<code>docker version</code>	Retrieves docker versioning

Image and Container Information

<code>docker ps</code>	List all running containers
<code>docker ps -a</code>	List all container instances, with their ID and status
<code>docker images</code>	Lists all images on the local machine
<code>docker history <user>/<i-image></code>	Lists the history of an image
<code>docker logs <container></code>	Displays the logs from a running container
<code>docker port <container></code>	Displays the exposed port of a running container
<code>docker diff <container></code>	Lists the changes made to a container
<code>docker logs [-f] <container></code>	Read container stdout
<code>docker top <container></code>	Show running processes
<code>docker inspect <container></code>	Show container configuration
<code>docker events</code>	Show container events

`<container>` can be replaced with container **name** or **ID**

Network Commands

<code>docker network ls</code>	List networks
<code>docker network inspect <Network></code>	Inspect a network
<code>docker network create <Network></code>	Create a network
<code>docker network rm <Network></code>	Removes a network
<code>docker network connect <Network> <Container></code>	Connect a container to the network
<code>docker network connect --ip <IP> <Network> <Container></code>	Specify the IP address of container interface
<code>docker network disconnect <Network-Name> <Container></code>	Disconnect container from network

`<Network>` can be replaced with the network **ID** or **NAME**

Work With Images and Containers

<code>docker run -it user/image</code>	Runs an image, creating a container and changing the terminal to the terminal within the container
<code>docker run -p <hostport>:<containerport> -d user/image</code>	Run an image in detached mode with port forwarding
<code>docker start <container></code>	Start a container
<code>docker exec <container> [shell command]</code>	Executes a command within a running container, e.g. <code>/bin/bash</code>
<code>docker attach <container></code>	Changes the command prompt from the host to a running container
<code>docker stop <container></code>	Stop a container
<code>docker rm -f <container></code>	Delete a container
<code>docker create <image>: <tag></code>	Create image
<code>docker rmi <image></code>	Delete an image
<code>docker tag <user>/<i-image>:<tag> <user>/<i-image>:<new tag></code>	Add a new tag to an image
<code>ctrl+p</code> then <code>ctrl+q</code>	Changes the command prompt from the host to a running container

`<container>` can be replaced with container **name** or **ID**



Misc. Examples

<code>sudo docker commit <i>containerID</i> username/image</code>	Commit from container to an image
<code>docker push username/image</code>	Docker push example (after logging)
<code>docker run -it -v /host/directory:/container/directory [options] <i>image</i> [command]</code>	Mount a host directory in a Docker container
<code>docker cp /host/directory/file.ext <i>container:/container/directory</i></code>	Copy file from host machine to docker container
<code>docker rmi \$(docker images -a -q)</code>	Delete all Docker images
<code>docker rm \$(sudo docker ps -a -q)</code>	Delete all stopped containers
<code>docker rm \$(sudo docker ps -a grep Exit cut -d ' ' -f 1)</code>	Delete all containers in Exit state
<code>docker save <i>image</i> > <i>filename.tar</i></code>	Export image to a file
<code>docker load < <i>filename.tar</i></code>	Load image from file
<code>docker [command] --help</code>	Command help

Options Keywords

<code>-p, --publish</code>	Host to container port mapping
<code>--publish-all</code>	Publish all ports
<code>-d, --detach</code>	Run in background
<code>-e, --env</code>	Set env vars
<code>-v, --volume</code>	Mount files or directories



By **angel_ceed**
cheatography.com/angel-ceed/

Published 8th May, 2019.
 Last updated 10th May, 2019.
 Page 2 of 2.

Sponsored by **Readable.com**
 Measure your website readability!
<https://readable.com>