

### Basic Docker commands

<code>docker --help</code>	Displays basic help options and commands
<code>docker &lt;command&gt; --help</code>	Display detailed help for a specific command
<code>docker ps</code>	List of "live" containers
<code>docker ps -a/-all</code>	List of all containers
<code>docker version</code>	Display docker versioning

### Container commands

<code>docker run &lt;image&gt;</code>	Builds a docker image from the Docker file
<code>docker run [options] &lt;image:tag@digest&gt; &lt;command&gt; &lt;arguments&gt;</code>	Full syntax of docker run command
<code>docker create &lt;image&gt;:&lt;tag&gt;</code>	Creates a container using the image
<code>docker start &lt;container&gt;</code>	Starts a container
<code>docker stop &lt;container&gt;</code>	Stops a container in an orderly manner
<code>docker kill &lt;container&gt;</code>	Immediately stops a running container
<code>docker restart &lt;container&gt;</code>	Restarts a stopped container
<code>docker rm &lt;container&gt;</code>	Removes a stopped container
<code>docker port &lt;container&gt;</code>	Lists the port mapping of a "running" container

`<container>` can be replaced with the containers unique **ID** or unique **NAME**  
`<tag>` is optional and will default to *latest*

### Image Commands

<code>docker images</code>	List images
<code>docker run &lt;image&gt;:&lt;tag&gt;</code>	Builds a new container using the image
<code>docker build &lt;image&gt;:&lt;tag&gt;</code>	Builds a docker image from the Docker file
<code>docker rmi &lt;image&gt;:&lt;tag&gt;</code>	Remove an image (not attached to a container)
<code>docker save &lt;image&gt;:&lt;tag&gt;</code>	Saves an image to a <i>tar</i> archive
<code>docker push &lt;image&gt;:&lt;tag&gt;</code>	Pushes the image to Docker Hub

### Image Commands (cont)

<code>docker pull &lt;image&gt;:&lt;tag&gt;</code>	Pulls the image from Docker Hub
--	---------------------------------

`<image>` can be replaced with the images unique **ID** or or unique **NAME**  
`<tag>` is optional and will default to *latest*

### Miscellaneous Commands

<code>docker exec -it &lt;container&gt;</code>	Executes the command or terminal of a running container
<code>docker inspect &lt;image or container&gt;</code>	Returns low level information on an image or a container
<code>docker attach &lt;container&gt;</code>	Attach to a running container

### Option Keywords

<code>-i, --interactive</code>	Keep STDIN open even if not attached
<code>-t, --tty</code>	Allocate a psuedo-TTY
<code>-it</code>	Combines the <code>-i</code> and the <code>-t</code> options (interactive response)
<code>-a, --attach</code>	attach
<code>-d, --detach</code>	Run in background
<code>-e, --env</code>	Set ENV variables
<code>exec</code>	Run a new command in a container (See Misc)
<code>-p, -publish &lt;host_port&gt;:&lt;container_port&gt;</code>	Port mapping (Host to container)
<code>-v, --volume &lt;source&gt;:&lt;target&gt;</code>	Mount files or directories (terse)
<code>-m, --mount</code>	Mount files or directoroes (verbose)

### Arguments

<code>--name=&lt;name&gt;</code>	Assigns a name to a container: <code>docker run --name=&lt;name&gt; &lt;image&gt;</code>
<code>-a, --all</code>	Adds the option of "all" to the container list: <code>docker ps -a</code>
<code>sh</code>	Entrypoint command for a <b>shell</b> in a container: <code>docker run -it &lt;image&gt; sh</code> or <code>docker exec -it &lt;container&gt; sh</code>



---

By **Cash** (CashM)  
[cheatography.com/cashm/](https://cheatography.com/cashm/)

Published 23rd January, 2022.  
Last updated 28th January, 2022.  
Page 1 of 2.

---

Sponsored by **ApolloPad.com**  
Everyone has a novel in them. Finish  
Yours!  
<https://apollopad.com>

### Volume Commands

docker volume ls	List volumes
docker volume inspect	Inspect a volume
docker volume create <volume>	Create a volume
docker volume rm <volume>	Removes a volume

<volume> can be replaced with the volumes unique **ID** or unique **NAME**

### Dockerfile (for creating images)

FROM <image> [AS <name>]	Sets the base image for this layered imaged
RUN <command>	(Shell form) Executes commands in a new layer
RUN ["executable", "param1", "param2"]	(Exec form) Executes commands in a new layer
CMD ["executable", "param1", "param2"]	(Exec form) Sets default executable as the ENTRYPOINT
CMD ["param1", "param2"]	(adds default parameters to non-specified ENTRYPOINT)
CMD command param1 param2	(shell form)
LABEL <key>=<value> <key>=<value> ...	Adds metadata to an image
EXPOSE <port> [<ports>/<protocol>]	Specifies the port(s) for the container to listen on
ENV <key>=<value>	Sets an environment variable for all subsequent instructions
COPY [--chown=<usr>:<group>] <src>... <dest>	Copies files or directories from <src> and adds them to the container at <dest>
VOLUME ["/data"]	(Can be an array) Creates a mount point for files or directories
USER <user>[:<group>]	Sets the user name and optionally the user group to use when running image
WORKDIR	Sets the working directory
ARG <NAME>=<value>	ARG is the only command that may precede the FROM command. It sets support variables for the FROM

### Basic Dockerfile Dev Structure

From <image>
WORKDIR <path>
[ENV <key>=<value>]
COPY <src> <target>
RUN command param1 param2 ...
CMD ["executable", "param1", "param2", ...]

