Docker basics Cheat Sheet

by FooFax (foobar) via cheatography.com/16755/cs/35110/

-

Glossary

Build	build is the process of building Docker images using a Dockerfile.
Container	a container is a runtime instance of a Docker image.
Dockerfile	a Dockerfile is a text document that contains all the commands you would normally execute manually in order to build a Docker image. Docker can build images automatically by reading the instructions from a Dockerfile.
Docker Daemon	the background service running on the host that manages building, running and distributing Docker containers. The daemon is the process that runs in the operating system which clients talk to.
Image	a Docker image is a read-only template that contains a set of instructions for creating a container.
Stack	a group of containers that runs together in order to create a service. Example: [wordpress] + [mySql]
Volume	Docker volumes are file systems mounted into a Docker containers to preserve data generated by the running container.

Author this cheat sheet has been created by TheFax

Run containers

docker run	`parameter`	`descri ption`
	-d	Run in background / detached mode
	-it	Interactive, tty
	--rm	Remove container after it stops.
	-p ***83:80***	Port mapping host:container
	-v ***/dirHost:/dir***	Mount file or directories /host/dir:/container/dir
	--name ***pluto***	Set container name
	--restart ***no***	Restart policy: **no**: stops the container when it exit. **on-failure**[:max-retries]: Restart only if the container exits with a non-zero exit status. Optionally, limit the number of restart retries. **always**: restart the container regardless of the exit status. The container will also always start on daemon startup. **unless-stopped** : Always restart the container regardless of the exit status, except if the container was put into a stopped state.
	IMAGE	Image name is the LAST parameter.

Manage containers

docker start/stop ***CONTAINER***	Start or stop a container
docker pause/unpause ***CONTAINER***	Pause or unpause a container

Manage containers (cont)	
docker restart *CONTAINER*	Stop and restart a container
docker attach *CONTAINER*	Connect terminal to a running container
docker rename *OLD_NAME NEW_NAME*	Rename a container
docker container rm *CONTAINER* **docker rm** *CONTAINER*	Remove the specified container
docker container rm -f *CONTAINER* **docker rm -f** *CONTAINER*	Force the removing of specified container.
docker container prune	Delete all stopped containers

Manage images	
docker rmi *IMAGE*, or **docker image rm** *IMAGE*	Remove image
docker build *URL*	Create an image from a dockerfile
docker commit *CONTAINER NEW_IMAGE_NAME*	Create an image from a container
docker search *TERM*	Search the Docker Hub for images

Information	
docker stats	Live data about running containers
docker ps	List running containers
docker ps -a	List ALL containers, also stopped ones
docker port *CONTAINER*	List port mapping of specified container
docker image ls, or **docker images**	List all locally stored images
docker container ls	List running containers
docker container ls -a	List all containers
docker volume ls	List volumes

Tips :-)	
Where are docker images and files?	/var/lib/docker
How can I refer to a container?	Via its name (set by *--name Pluto*) or... Via its ID (example: 4a7f7eebae0f63178aff7eb0aa39f0627a203ab2df258c1a00b456cf20063) or... Via the first letters of its ID (example: 4a7f)
How can I test my docker?	Try one of these commands: **docker run hello-world** **docker run -it --rm -d --name test -p 83:80 nginx**