Cheatography

Docker Tools and Terms Cheat Sheet

by achang via cheatography.com/39709/cs/12310/

Container Tools

`docker ps`	lists all currently running containers
`docker ps -a`	lists all containers (including stopped ones)
`docker logs CONTAINER(S)`	shows the output of the given container(s)
`docker rm CONTAINER(S)`	removes a container(s) - including any local data changes
`docker rm -f CONTAINER(S)`	same as `docker rm` except also force stops a container(s)
`docker start CONTAINER(S)`	attempts to start a stopped container(s)
`docker stop CONTAINER(S)`	attempts to stop a running container(s)
`docker restart CONTAINER(S)`	stops and then starts the specified container(s)
`docker cp SRC DEST`	copies files to/from a specific container to a local location container location format: `CONTAINER:PATH_ON_CONTAINER`
`docker exec -it CONTAINER bash`	starts a bash terminal session in the CONTAINER specified

Aliases

`docker_build`	`docker_compose` followed by a `docker_build_js`
`docker_build_js`	builds front-end components using `yarn run compile`
`docker_compose`	installs dependencies as defined in `composer.lock` - does not update
`docker_start`	runs all containers as defined in `docker-compose.yml` with the default command as defined by the image
`dockerize`	logs in to aws in order to pull new images
`src_sync`	copies source code into all running containers and makes sure container source is up to date
`docker_cache_clear`	runs `CacheClearCommand` in order to remove cached assets
`docker_cache_force_clear`	`rm -rf app/cache`
`docker_run_single_test TEST_FILE_LOCATION`	runs a single phpUnit test file
`cmd`	starts a new container using the scriptbox image and starts an interactive bash session

By **achang**
cheatography.com/achang/

Published 17th July, 2017.
Last updated 29th August, 2017.

Cheatography

Docker Tools and Terms Cheat Sheet

by achang via cheatography.com/39709/cs/12310/

Terms

Image	A snapshot of a lightweight filesystem in order to run specific tasks
Container	An instantiation of an environment created from a specific image
Filecabinet	A file mount that is shared across all containers

Troubleshooting

`docker ps -a`	use to see which containers are not starting if linking errors occur
`docker logs CONTAINER`	use to see specific logs and errors for a container to trouble shoot bad exit codes
`src_sync`	use if logs reveal that files cannot be found
`docker_build/build_js/compose`	use if parameters/dependencies seem to be incorrect
`docker rm CONTAINER`	use if a file is corrupt in a container NOTE: DO NOT REMOVE `*_data` CONTAINERS AS THEY WILL TAKE A LONG TIME TO DOWNLOAD
`docker_cache_clear/docker_cache_force_clear`	use if annotations or other cached assets seem to be corrupt/not updating

Docker ps columns

NAMES	Interchangeable with `CONTAINER ID` - used to reference a specific container
CONTAINER ID	Interchangeable with `NAMES` - used to reference a specific container
IMAGE	Image that the container environment is running on
COMMAND	Command(s) that the container is runs when started
CREATED	Date when container was first created
STATUS	Current status of container - will also display exit codes
PORTS	Displays external -> internal port mapping and protocal

xdebug - docker native

`sudo ifconfig lo0 alias 10.254.254.254`

`docker exec -it webserver bash`

`vi /etc/php.d/xdebug.ini`

press `i` to be able to edit, arrows to navigate

set `xdebug.remote_connect_back=0`

set `xdebug.remote_host=10.254.254.254`

set `xdebug.remote_autostart=1`

to save and quit from vi:

press `esc`

type `:wq`

press `enter`

`exit` to get out of the container

run `docker restart webserver` in order to have the new config values take place

By **achang**
cheatography.com/achang/

Published 17th July, 2017.
Last updated 29th August, 2017.