Learning docker 01: Basics

**### Verify docker install**

$ docker run hello-world

**### Run an Ubuntu container in the foreground**

$ docker run –it ubuntu bash

-t : allocate a pseudo—tty

-i : keep STDIN open even if not attached

**### Run an nginx container in background mode and mapping a port**

$ docker run –d –p 8080:80 nginx

-d=False : detached mode

-p=[] : port mapping – publish container’s port or range or ports to host

format: ip:hostPort:containerPort | ip::containerPort

Learning docker 02: Building images

### Tree

| Dockerfile

| html/

| ---------index.html

### Dockerfile

FROM nginx

COPY html /usr/share/nginx/html

### Command

$ docker build –tag my\_http .

$ docker run -d -p 8080:80 --name my\_http\_container my\_http

Learning docker 03: Attaching volumes

Starting point – no volumes

Dockerfile

FROM php:7.2-cli

COPY ./src /usr/src/myapp <-- this will be burned into the image / be made immutable

WORKDIR /usr/src/myapp

CMD ["php", "./run.php"]

run.php

<?php

echo 'Running in docker';

$ docker build –t run-php .

$ winpty docker run -it --rm --name run-app run-php

\* changing the content of run.php won’t be reflected until the image is re-built

Attaching volumes

$ docker run -it --rm --name php-runner -v //c/Users/ronaldg/Desktop/learning/docker/tutorials/willitscale/tut3/src:/usr/src/myapp -w /usr/src/myapp php:7.2-cli php run.php

-w : working dir

docker run -it --rm --name php-runner –v //c:/users/ronaldg/Desktop/learning/docker/tutorials/willitscale/tut3/src:/usr/src/myapp -w /usr/src/myapp php:7.2-cli php run.php

\* doesn’t work, figuring out why

Learning docker 04: Compose

Dockerfile

FROM php:7.2-apache

docker-compose.yml

version: "3"

services:

www:

build: .

ports:

- "8040:80"

volumes:

- ./src:/var/www/html/

networks:

- default

/src/index.php

<?php

echo "We're in our composed containers";

Learning docker 05: Customising Docker Images with Builder