Docker Lab 2

==========================

1. Running more containers

===========================

Now that you have seen how docker commands work and how to use the docker run command to run containers, lets start some more containers for fun.

#docker run -it ubuntu

#docker run -it httpd

#docker run -itd httpd ( d - detached mode, httpd container will start running as a deamon)

#docker run -it java

#docker run -it python

Check the running containers

# docker ps

2. Accessing the httpd docker containers using host ports

===========================================================

Start the docker httpd container with a port defined

#docker run -itd --name website1 -p 8080:80 httpd

#docker run -itd --name website2 -p 8081:80 httpd

use the docker ps command and make a note of the port mapping

#docker ps

We started 2 httpd containers in the same server. The containers are using the port 80 to run the httpd service, however, they are exposed to 8080 and 8081 ports in the host. We can access them using the below command

curl [http://<host](http://%3Chost/) ip address>:8080

curl [http://<host](http://%3Chost/) ip address>:8081

or open the browser and try to access them. in case you are using amazon EC2 instances, use the public ip address

3. Dynamic mapping of ports

==============================

We can use dynamic port mapping instead of manual port mapping

#docker run -itd --name website -P httpd

-P (capital P, allows the host to allocate a free port on it's own)