[Docker Tutorials --- From Youtube](https://www.youtube.com/playlist?list=PLkA60AVN3hh_6cAz8TUGtkYbJSL2bdZ4h)

gist.github.com/botchagalupe

https://gist.github.com/botchagalupe/53695f50eebbd3eaa9aa

# Docker

Docker CommandLine: <https://docs.docker.com/engine/reference/commandline/cli/>

## Ubuntu

sudo apt-get install –y docker.io sudo usermod –aG docker vagrant

docker –v wget –q0- <https://get.docker.com/> | sh

sudo service docker start sudo chkconfig docker on

To run <<docker run –it ubuntu bash>>

sudo su – service docker start

docker images sudo usermod –aG docker a

docker run –it Ubuntu bash

sudo docker ps sudo docker ps –a

docker run –i busybox: run in the container

ctrl–D ctrl–Q

tty sudo docker run –t busybox docker ps

interactive docker run –it busybox interactive mode ls / pwd / hostname

detach docker run –d busybox detach mode

docker run –itd busybox docker ps

detach docker attach containerID detach docker: Ctrl+p + Ctrl+q

long name docker ps --no-trunc=true docker ps -a --no-trunc=true

cid=$(docker run -itd busybox) echo $cid

docker inspect $cid docker stop dockerID docker stop $cid

docker run --name john1 -itd busybox docker stop john1

kill docker ps –q docker kill $(docker ps -q)

remove rm john1 docker rm $(docker ps -aq)

### [Docker Tutorial 3 - Fun With Volumes](https://www.youtube.com/playlist?list=PLkA60AVN3hh_6cAz8TUGtkYbJSL2bdZ4h)

[Data volumns](https://docs.docker.com/engine/tutorials/dockervolumes/) layers

docker run -it -v /john1 busybox cd john1 touch file1 Ctrl-p-q

docker restart <cid> docker exec <cid> ls /john1

list images docker images docker inspect busybox

docker history

docker fun -it -v /john1 myimage touch file3 file4 file5

docker run -it -v /home/a/john3:/john3 busybox

docker run -it -v /home/a/john3:/john3:ro busybox

docker run -it -v ~/.bash\_history:/.bash\_history busybox

docker kill $(docker ps -q)

docker rm $(docker ps -aq)

docker run -it --name john1 -v /john1 busybox

docker run -it --name john2 --volumes-from john1 busybox

docker run -it --name john3 --volumes-from john2 busybox

docker run --rm --volumes-from john1 -v $(pwd):/backup busybox tar cvf /backup/john2.tar /john ???

# [**Docker Tutorial 4 - More Run With a Couple of Extra Goodies**](https://www.youtube.com/watch?v=ZgjHM1XmAKI&index=4&list=PLkA60AVN3hh_6cAz8TUGtkYbJSL2bdZ4h)

docker -v docker search ubuntu docker search -s 1 ubuntu

docker images docker pull ubuntu docker pull:14.04 docker pull ubuntu:trusty

docker run -itd ubuntu docker images docker history <image id>

docker run -itd ubuntu /bin/sh docker run -itd ubuntu uname -a docker ps -a

docker logs <docker id> docker run -itd ubuntu sleep 10 && watch docker ps

docker kill $(docker ps -q) && docker rm $(docker ps -aq)

docker run -itd --name job1 ubuntu /bin/sh -c "while true; do echo Docker Rocks; sleep 1; done"

tail ???

docker kill $(docker ps -lq) kill the last one

docker rm $(docker ps -lq) kill the last one

docker run -itd --name job2 ubuntu /bin/sh -c "while true; do echo Docker Rocks; sleep 1; done"

docker stats job2

watch docker top job2 -ef

docker exec -itd job2 sleep 20

docker inspect -h

docker inspect --format '{{.Name}} {{.State.Running}}' job2 docker ps -a

docker run -itd --name job4 --label=NodeNumber=3 --label=NodeType=cluster ubuntu

docker inspect job4 > jmw.out

docker inspect --format '{{.Name}} {{.Config.Labels.NodeType}}' job4

cid=$(docker run -itd --ulimit nofile=1024:1024 ubuntu) ???

docker attach $cid ???

# [**Docker Tutorial 6 - Dockerfiles (Part 1)**](https://www.youtube.com/watch?v=gG_x28rDxus&list=PLkA60AVN3hh_6cAz8TUGtkYbJSL2bdZ4h&index=5)

basic three commands FROM, RUN, CMD

apache-ex1

FROM ubuntu:14.04

RUN apt-get -y install apache2

CMD ["/usr/sbin/apache2ctl", "-D", "FOREGROUND"]

Commands

docker build -f apache-dockerfile-ex1 -t apache-ex1 .

cid=$(docker run -itd apache-ex1)

docker exec $cid ip a

curl 172.17.0.179

nid=$(docker inspect --format '{{.NetworkSettings.IPAddress}}' $cid)

curl $nid

Caching

docker build -f apache-dockerfile-ex1 -t apache-ex1 .

docker build --no-cache=true -f apache-dockerfile-ex1 -t apache-ex1 .

alternative syntax

apache-ex2

From ubuntu:14.04

RUN ["apt-get", "install", "-y", "apache2"]

CMD /usr/sbin/apache2ctl -D FOREGROUND

Finding index.html

docker run -it apache-ex1 /bin/sh

find / --name index.html

More commands ADD and EXPOSE

apache-ex3

FROM ubuntu:14.04

RUN \

apt-get update && \

apt-get -y install apache2

ADD index.html /var/www/html/index.html

EXPOSE 80

CMD ["/usr/sbin/apache2ctl", "-D", "FOREGROUND"]

index.html

<h1>Docker Rock1!</h1>

Commands:

cat index.html

docker build -f apache-dockerfile-ex3 -t apache-ex3 .

cid=$(docker run itd -P apache-ex3)

nid=$(docker inspect --format "{{.NetworkSettings.IPAddress}}' $cid)

curl $nid

ip a show eth1

apache-ex4

FROM ubuntu:14.04

VOLUME ["/var/www/html"]

RUN \

apt-get update && \

apt-get -y install apache2

EXPOSE 80

CMD ["/usr/sbin/apache2ctl", "-D", "FOREGROUND"]

Commands

docker build -f apache-dockerfile-ex4 -t apache-ex4 .

cid=$(docker run -itd -v ~/test/index.html:/var/www/html/index.html wordpress-ex4)

nid=$(docker inspect --format '{{.NetworkSettings.IPAdress}}' $cid)

curl $nid

# [**Docker Tutorial 7 - Dockerfiles (Part 2)**](https://www.youtube.com/watch?v=L6bjTlVdc6U&list=PLkA60AVN3hh_6cAz8TUGtkYbJSL2bdZ4h&index=6)

git clone https://github.com/docker-library/wordpress.git

cd ./m/wordpress/ docker build -t docker-library/wordpress .

docker run -d -p 80:80 -p 3306:3306 tutum/lamp

https://github.com/tutumcloud/lamp

git clone https://github.com/tutumcloud/tutum-docker-lamp.git

... ... ... ... ...

https://github.com/docker/docker

# [**Docker Tutorial 8 - Troubleshooting with Sysdig**](https://www.youtube.com/watch?v=S-fYCOMnsbY&list=PLkA60AVN3hh_6cAz8TUGtkYbJSL2bdZ4h&index=7)

[Let There Be Light – Sysdig Adds Container Visibility](https://sysdig.com/blog/let-light-sysdig-adds-container-visibility/)

$ sudo docker run --name mysql -e MYSQL\_ROOT\_PASSWORD='password' -d mysql

$ sudo docker run --name wordpress1 --link mysql:mysql -d wordpress

$ sudo docker run --name wordpress2 --link mysql:mysql -d wordpress

$ sudo docker run --name wordpress3 --link mysql:mysql -d wordpress

$ sudo docker run --name wordpress4 --link mysql:mysql -d wordpress

$ sudo docker run --name haproxy -p 80:80 --link wordpress1:wordpress1 --link wordpress2:wordpress2 --link wordpress3:wordpress3 --link wordpress4:wordpress4 -d tutum/haproxy

www.github.com/docker-library/wordpress/tree/master/apache

ps -e9 user,pid,start\_time,ucmd,cgroup

curl -s https://s3.amazonaws.com/download.draios.com/stable/install-sysdig | sudo bash

sudo sysdig -c lscontainers

sudo sysdig -c topcontainers\_cpu

sudo sysdig -pc -c topcontainers\_cpu

sudo sysdig -cl

docker top mysql aux

docker stats mysql

sudo sysdig -pc -c topprocs\_cpu

sudo syddig -pc -c topprocs\_cpu container.name contains job1

sudo sysdig -pc -c topcontainers\_net

while true; do curl localhost; sleep 1; done

sudo sysdig -pc -c topprocs\_net

[**Docker Tutorial 9 - Docker Machine**](https://www.youtube.com/watch?v=3zXe3uXQnSY&list=PLkA60AVN3hh_6cAz8TUGtkYbJSL2bdZ4h&index=8)

[Docker machine](https://docs.docker.com/machine/)

[docker machine installation from github](https://github.com/docker/machine/releases/tag/v0.8.0)

curl -L https://github.com/docker/machine/releases/download/v0.8.0/docker-machine-`uname -s`-`uname -m` >/usr/local/bin/docker-machine && chmod +x /usr/local/bin/docker-machine

https://github.com/docker/machine/releases/download/v0.8.0/docker-machine-Linux-x86\_64

Docker Tutorial 1 - Installing Docker

Docker Tutorial 2 - Docker Run Command Basics

Docker Tutorial 3 - Fun With Volumes

Docker Tutorial 4 - More Run With a Couple of Extra Goodies

Docker Tutorial 5 - Basic Networking

Docker Tutorial 6 - Dockerfiles (Part 1)

Docker Tutorial 7 - Dockerfiles (Part 2)

Docker Tutorial 8 - Troubleshooting with Sysdig

Docker Tutorial 9 - Docker Machine

Docker Tutorial 10 - Docker Compose

Docker Tutorial 11 - Docker Swarm

Docker Tutorial 12 - Kitematic

Docker Tutorial 13 - Getting Started with the new experimental Docker Networking

docker-machine -v

docker-machine create --driver-virtalbox vbox-test

docker-machine ls

docker-machine env dev1

eval "$(docker-machine env dev1)"

docker run busybox echo hello world

docker-machine create --driver digitalocean --digitalocean-access-token=$DIGITAL\_OCEAN\_TOKEN dev2

docker-machine ls **https://docs.docker.com/machine/#digital-ocean**

eval "$(docker-machine env dev2)"

docker-machine ls

docker run busybox echo hello world

https://docs.docker.com/machine/#amazon-web-services

docker-machine -D create --driver amazonec2 --amazonec2-access-key $AWS\_ACCESS\_KEY\_ID --amazonwec2-secret-key $AWS\_SECRET\_ACCESS\_KEY --amazonec2-vpc-id $AWS\_VPC\_ID --amazonec2-zone b dev3

docker-machine-Linux-x86\_64

docker-machine stop dev3

[**Docker Tutorial 10 - Docker Compose**](https://www.youtube.com/watch?v=gtoT0By8yh4&list=PLkA60AVN3hh_6cAz8TUGtkYbJSL2bdZ4h&index=9)