```
apt-get update -y
sudo apt-get install apt-transport-https ca-certificates curl
software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key
add -
sudo apt-key fingerprint 0EBFCD88
sudo add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
sudo apt-get update
sudo apt-get install docker-ce
```

```
docker -v
docker info

docker pull fedora
docker images
docker info
docker run -id --name f21 docker.io/fedora bash
docker ps
docker exec -it containerid /bin/bash
```

https://docs.docker.com/engine/userguide

```
docker run --rm ubuntu ping google.com

docker run --rm ubuntu /bin/bash -c "apt-get update -y && apt-get
install iputils-ping -y && ping google.com"
```

1. Foreground

```
docker run ubuntu:14.04 /bin/echo 'Hello world'
docker images
docker ps
docker ps -a
```

2. interactive

```
docker run -t -i ubuntu:14.04 /bin/bash
```

3.Daemon Mode

```
docker run -d ubuntu:14.04 /bin/sh -c "while true; do echo hello
world; sleep 1; done"

docker exec -it containerid /bin/bash
docker logs containerid
```

```
docker rm $(docker ps -aq --filter "status=exited")
docker top containerid


---Wordpress Application

docker run --name test-mysql -e MYSQL_ROOT_PASSWORD=password -d mysql:latest

docker run --name test-wordpress -p 9200:80 --link test-mysql:mysql -d wordpress


docker run --rm ubuntu /bin/bash -c "apt-get update -y && apt-get install iputils-ping -y && ping google.com"

docker run -t -i ubuntu:14.04 /bin/bash

docker run --name test-mysql -e MYSQL_ROOT_PASSWORD=password -d mysql:latest

docker run --name test-wordpress -p 9200:80 --link test-mysql:mysql -d wordpress

http://collabedit.com/rb5dy

https://github.com/maheshkharwadkar/docker-k8s-lab/tree/master/code/docker/flask-hello-world

docker run --name test-pyton -p 8000:5000 -d ImageId

docker ps

docker exec -it containerid /bin/bash

mkdir -p /opt/docker-demo/flask-demo
cd /opt/docker-demo/flask-demo

--------Docker File Python Flask -------------------------
FROM python:2.7
COPY . /app
WORKDIR /app
RUN pip install -r requirements.txt
EXPOSE 5000
CMD [ "python", "app.py" ]

--------------------app.py---------------------------------
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello():
    return "Hello World!"

if __name__ == "__main__":
    app.run()
```

```
---------------------requirements.txt-----------------
Flask


docker build . -t pythondemo:latest
docker images
docker run --name test-python -d ImageId
docker exec -it container_id /bin/bash

curl 127.0.0.1:5000

docker ps

---------------Docker File Demo Nginx ----------------------
FROM fedora
MAINTAINER scollier <scollier@redhat.com>

RUN yum -y update && yum clean all
RUN yum -y install nginx && yum clean all
RUN echo "daemon off;" >> /etc/nginx/nginx.conf
RUN echo "nginx on Fedora" > /usr/share/nginx/html/index.html

EXPOSE 80

CMD [ "/usr/sbin/nginx" ]

-------------------------------------
docker build . -t nginxdemo:latest
docker run --name mynginx -p 5000:80 -d 5435e7421112
docker run --name mynginx --net=host -d 5435e7421112
curl localhost:5000



------------Creating images from running container_id
docker run -i -t fedora /bin/bash
yum install httpd

on New Window

docker images
docker commit -a "Mahesh Kharwadkar" -m "Fedora with HTTPD package"
445b0ea0cae2 mk/fedora:httpd


docker diff 445b0ea0cae2
docker exec <<new-Containerid>

----------------Bind Container port to the host
docker run -d  --name demo nginx
docker inspect --format {{.NetworkSettings.IPAddress}} demo
curl 172.17.0.2
docker run -d  -p 80:80 --name demo1 nginx
iptables -t nat -L -n

----------------------Create a new bridge network
docker network create -d bridge my-bridge
```

```
ip a
brctl show
apt-get install bridge-utils

docker run -d --name test1 --network my-bridge busybox sh -c "while true;do sleep 3600;done"

------------------------Host network deep dive
docker run -d --name test3 --net=host centos:7 /bin/bash -c "while true; do sleep 3600; done"

-----------------Docker-compose
curl -L https://github.com/docker/compose/releases/download/1.11.2/docker-compose-Linux-x86_64 -o /usr/local/bin/docker-compose''

sudo chmod +x /usr/local/bin/docker-compose

docker-compose -v

git clone https://github.com/maheshkharwadkar/example-voting-app

cd example-voting-app

docker-compose build

docker-compose up -d


------------------Docker Compose------------------------
mkdir -p /opt/docker-compose-demo
cd /opt/docker-compose-demo
git clone https://github.com/maheshkharwadkar/example-voting-app

cd example-voting-app

curl -L https://github.com/docker/compose/releases/download/1.11.2/docker-compose-Linux-x86_64 -o /usr/local/bin/docker-compose''

sudo chmod +x /usr/local/bin/docker-compose
docker-compose -v
docker-compose build
docker-compose up -d
docker-compose ps

--------chnages in docker-compose
environment:
  - POSTGRES_PASSWORD=password


---Cleanup----
docker-compose stop
docker-compose rm

docker network rm examplevotingapp_front-tier
docker network rm examplevotingapp_back-tier
```

```
docker-compose down --volumes
```