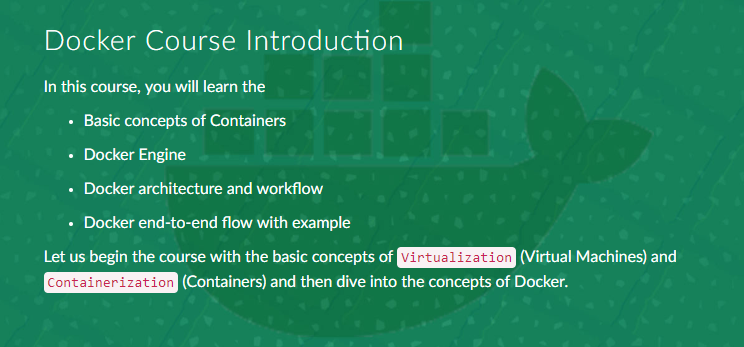
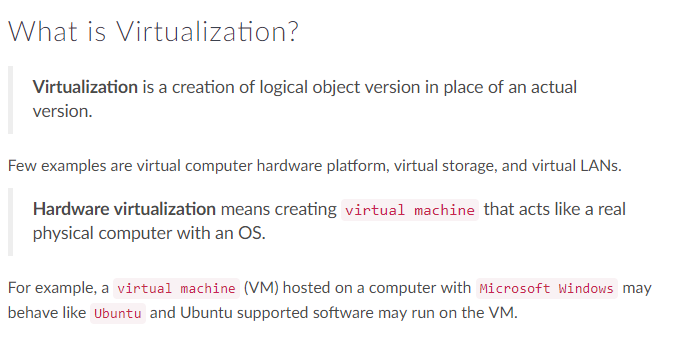
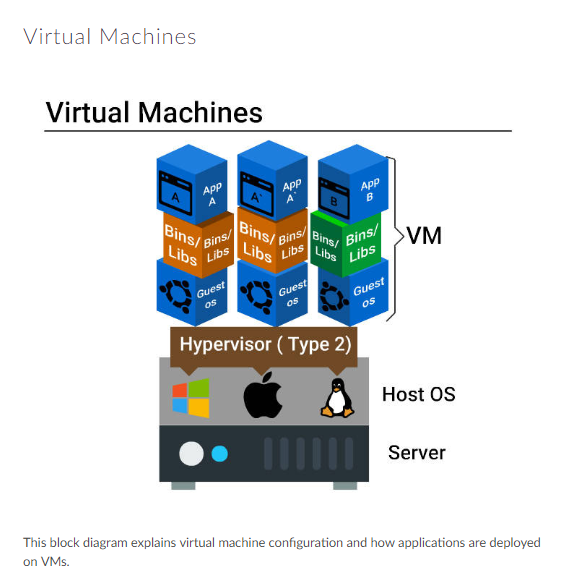
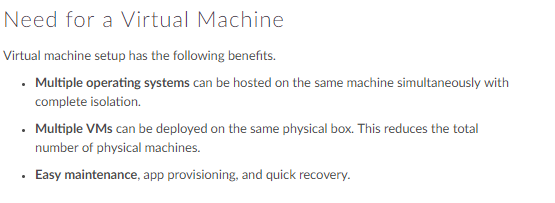
Docker

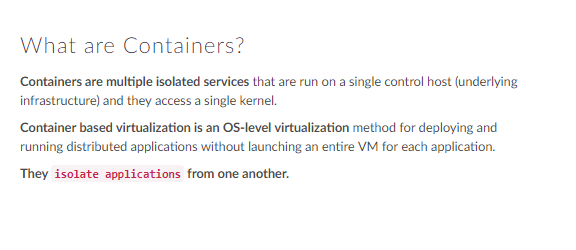


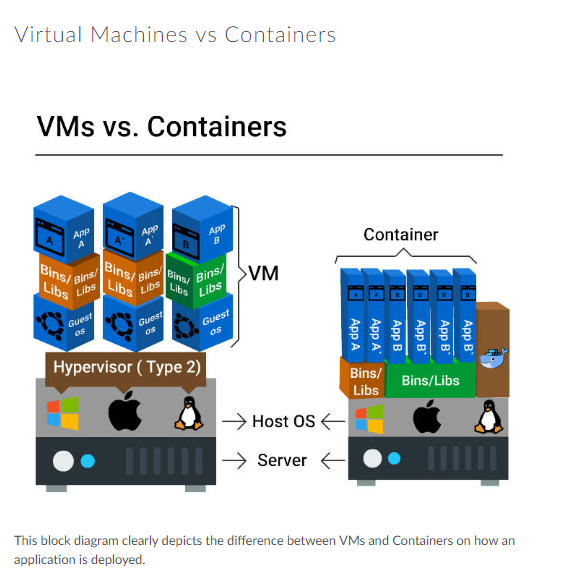


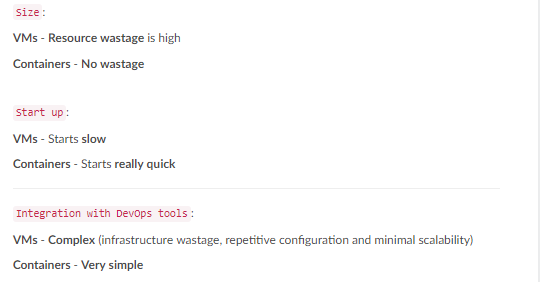




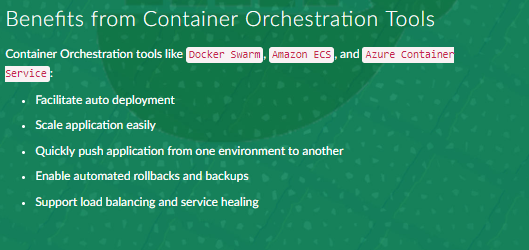


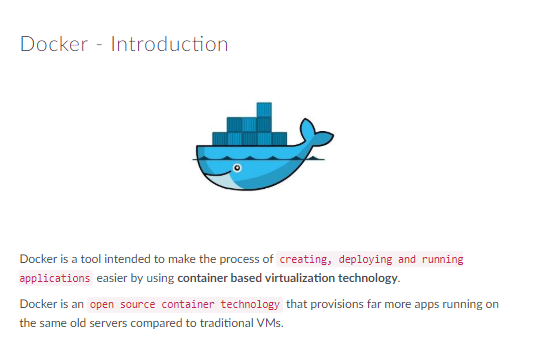


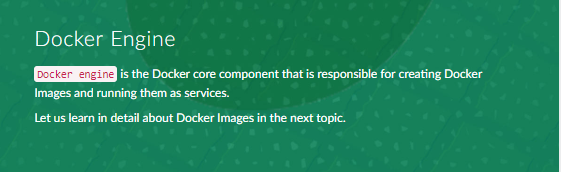


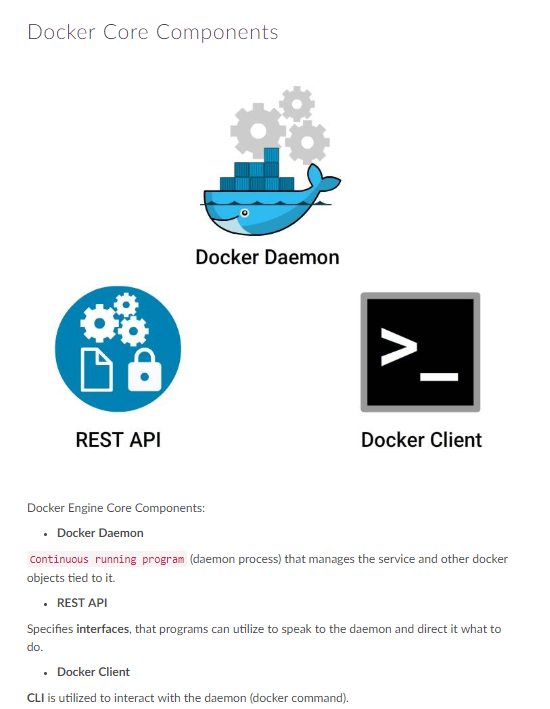


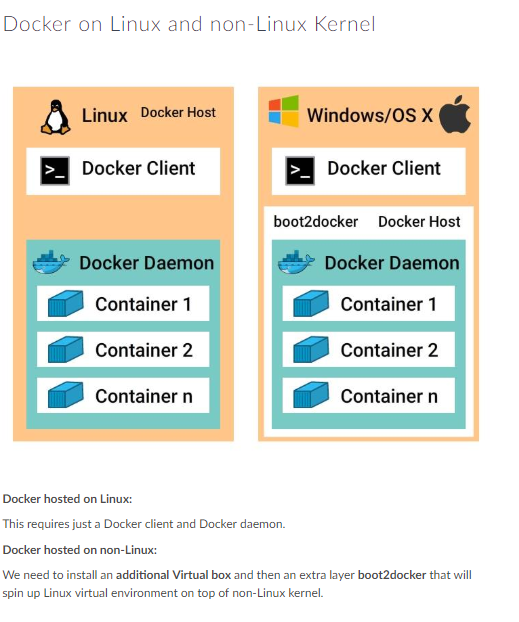


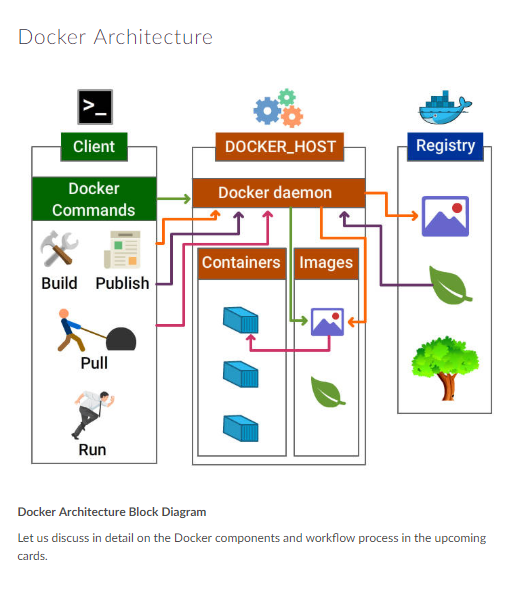




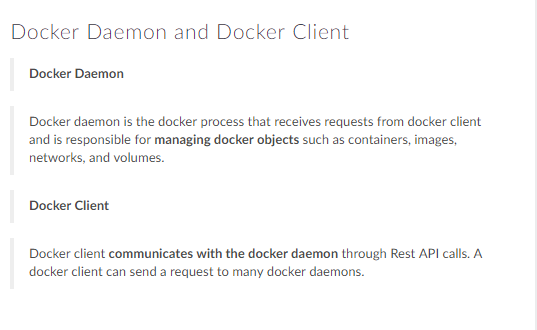


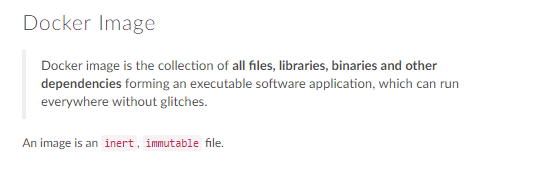


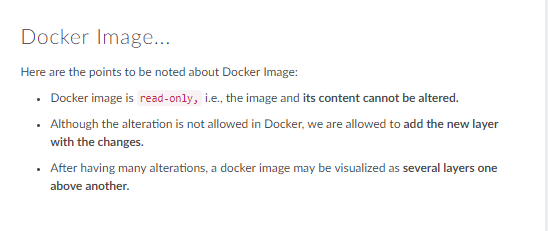


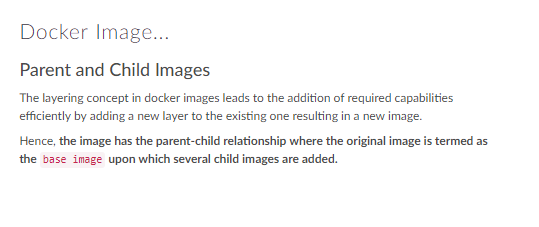


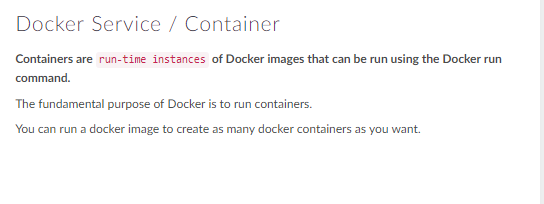


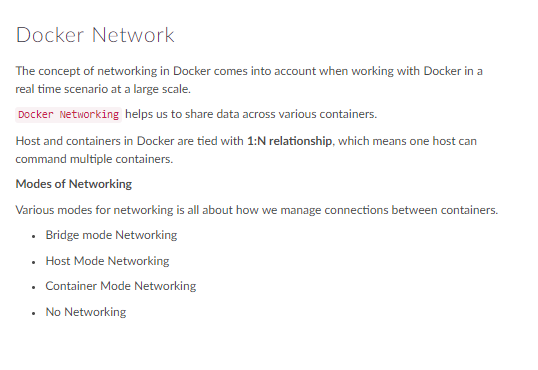


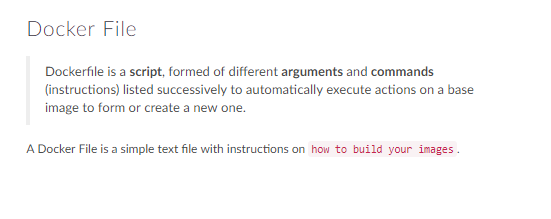


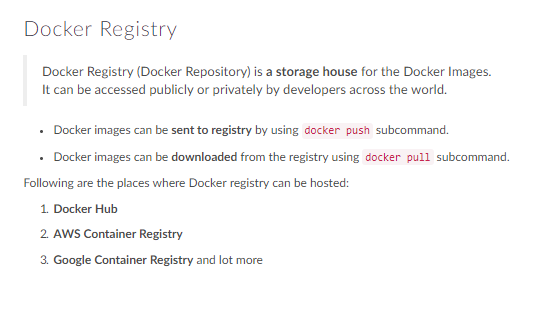


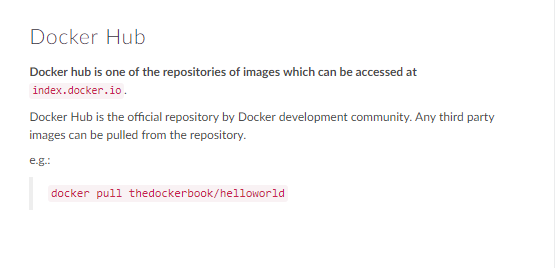


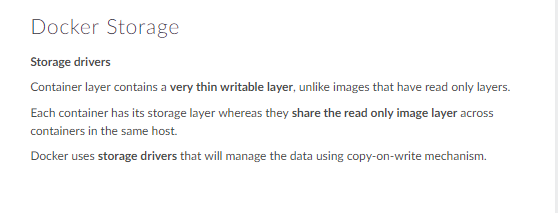


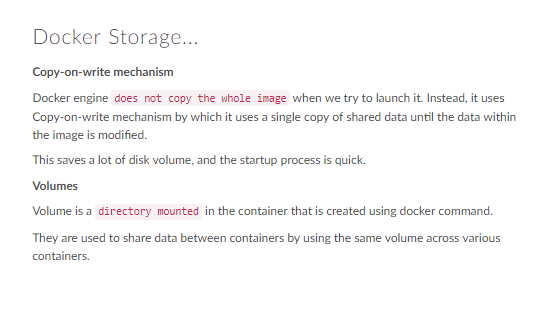


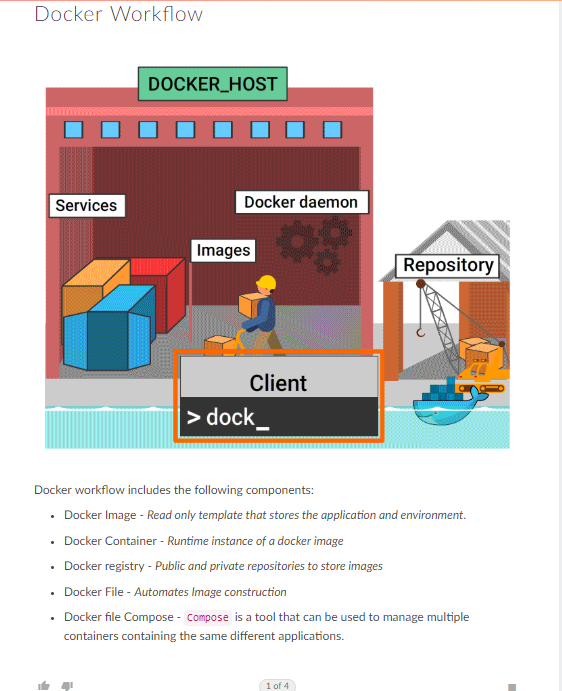


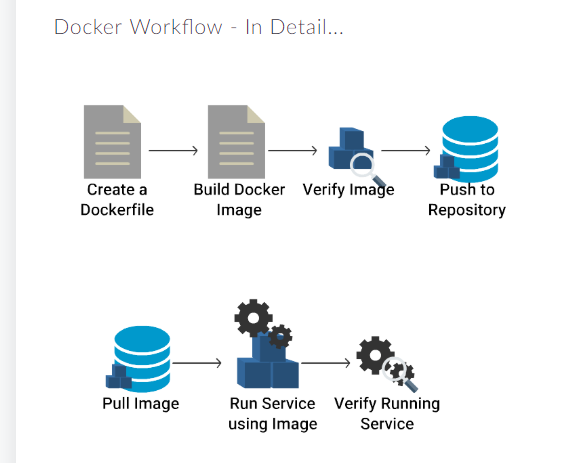












HANDS-ON:

Deploying your 1ST DOCKER container

$

$ docker search redis

NAME DESCRIPTION STARS OFFICIAL AUTOMATED

redis Redis is an open source key-value store that… 5631 [OK]

bitnami/redis Bitnami Redis Docker Image 87 [OK]

sameersbn/redis 71 [OK]

hypriot/rpi-redis Raspberry Pi compatible redis image 32

kubeguide/redis-master redis-master with "Hello World!" 26

kubeguide/guestbook-redis-slave Guestbook redis slave 20

redislabs/redis Clustered in-memory database engine compatib… 15

webhippie/redis Docker images for redis 9 [OK]

rediscommander/redis-commander Alpine image for redis-commander - Redis man… 8 [OK]

arm32v7/redis Redis is an open source key-value store that… 7

rtoma/logspout-redis-logstash Logspout including Redis adapter for sending… 5

oliver006/redis\_exporter Prometheus Exporter for Redis Metrics. Supp… 5

centos/redis-32-centos7 Redis in-memory data structure store, used a… 3

frodenas/redis A Docker Image for Redis 2 [OK]

dynomitedb/redis Redis backend for DynomiteDB. 2 [OK]

arm64v8/redis Redis is an open source key-value store that… 2

tomesar/redis-arm Redis for ARM! 2 [OK]

kilsoo75/redis-master This image is for the redis master of SK Clo… 1

tiredofit/redis Redis Server w/ Zabbix monitoring and S6 Ove… 1 [OK]

google/guestbook-python-redis A simple guestbook example written in Python… 1

circleci/redis CircleCI images for Redis 1 [OK]

wodby/redis Redis container image with orchestration 1 [OK]

anchorfree/redis redis cache server for logging 0

iadvize/redis 0

brendangibat/docker-logspout-redis Docker Logspout container with Logspout-Redi… 0 [OK]

$ docker run -d redis

330a642cfb1e2d6948106825a1475a0a23af4d55a8eb7d0e71c2e07f3d41ff4b

$ docker inspect redis

[

{

"Id": "sha256:4760dc956b2ddc9ac1c508936e39b63a22c6f0640ef58c1b10ff73f04e253ffe",

"RepoTags": [

"redis:latest"

],

"RepoDigests": [

"redis@sha256:26c93c5b06eaa323bb1089500f42b0dd158138772348b865e364127f1d554982"

],

"Parent": "",

"Comment": "",

"Created": "2018-03-14T19:11:50.675745683Z",

"Container": "8268efe7bcf96b627976a6fd49a39e0c002300f40bb13c490bff734bd9ab8912",

"ContainerConfig": {

"Hostname": "8268efe7bcf9",

"Domainname": "",

"User": "",

"AttachStdin": false,

"AttachStdout": false,

"AttachStderr": false,

"ExposedPorts": {

"6379/tcp": {}

},

"Tty": false,

"OpenStdin": false,

"StdinOnce": false,

"Env": [

"PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",

"GOSU\_VERSION=1.10",

"REDIS\_VERSION=4.0.8",

"REDIS\_DOWNLOAD\_URL=http://download.redis.io/releases/redis-4.0.8.tar.gz",

"REDIS\_DOWNLOAD\_SHA=ff0c38b8c156319249fec61e5018cf5b5fe63a65b61690bec798f4c998c232ad"

],

"Cmd": [

"/bin/sh",

"-c",

"#(nop) ",

"CMD [\"redis-server\"]"

],

"ArgsEscaped": true,

"Image": "sha256:f938399550833692dc52d1715ec2bf6084f17cc8557ee1053c1c393d69787d8e",

"Volumes": {

"/data": {}

},

"WorkingDir": "/data",

"Entrypoint": [

"docker-entrypoint.sh"

],

"OnBuild": [],

"Labels": {}

},

"DockerVersion": "17.06.2-ce",

"Author": "",

"Config": {

"Hostname": "",

"Domainname": "",

"User": "",

"AttachStdin": false,

"AttachStdout": false,

"AttachStderr": false,

"ExposedPorts": {

"6379/tcp": {}

},

"Tty": false,

"OpenStdin": false,

"StdinOnce": false,

"Env": [

"PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",

"GOSU\_VERSION=1.10",

"REDIS\_VERSION=4.0.8",

"REDIS\_DOWNLOAD\_URL=http://download.redis.io/releases/redis-4.0.8.tar.gz",

"REDIS\_DOWNLOAD\_SHA=ff0c38b8c156319249fec61e5018cf5b5fe63a65b61690bec798f4c998c232ad"

],

"Cmd": [

"redis-server"

],

"ArgsEscaped": true,

"Image": "sha256:f938399550833692dc52d1715ec2bf6084f17cc8557ee1053c1c393d69787d8e",

"Volumes": {

"/data": {}

},

"WorkingDir": "/data",

"Entrypoint": [

"docker-entrypoint.sh"

],

"OnBuild": [],

"Labels": null

},

"Architecture": "amd64",

"Os": "linux",

"Size": 106681215,

"VirtualSize": 106681215,

"GraphDriver": {

"Data": {

"RootDir": "/var/lib/docker/overlay/d5d206b9df9a3642afb5a4767df56cf1770db74b9de93724098e86acf3ff97f3/root"

},

"Name": "overlay"

},

"RootFS": {

"Type": "layers",

"Layers": [

"sha256:43efe85a991cac5894f91ee8f45b328bbacd14966d89a8a00b0d06060c64b5ad",

"sha256:27af3b14aa14e1a60cde6af2917f1b829d14261c6cbf24b00e69b39115a65bcd",

"sha256:b92d098685e5b6a78d215e7757265f9cb93792287b87f891de52e05d2ae71097",

"sha256:303f4ecc36fbadd0d8c80c3e8938067784b9a22f99f4770a01daad09913e49e0",

"sha256:bb5c6286e241532f34c666bea252f230f9c633402361a8d04943aac5fd98857b",

"sha256:0209c1ade8f43160424aa3f175e20eabc1fb0ae8749aa9bd6caf4108a3afdd6a"

]

},

"Metadata": {

"LastTagTime": "0001-01-01T00:00:00Z"

}

}

]

$ docker logs redis

Error: No such container: redis

$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

330a642cfb1e redis "docker-entrypoint.s…" About a minute ago Up About a minute 6379/tcp upbeat\_davinci

$ docker logs 330a642cfb1e

1:C 22 Aug 12:19:21.937 # oO0OoO0OoO0Oo Redis is starting oO0OoO0OoO0Oo

1:C 22 Aug 12:19:21.938 # Redis version=4.0.8, bits=64, commit=00000000, modified=0, pid=1, just started

1:C 22 Aug 12:19:21.938 # Warning: no config file specified, using the default config. In order to specify a config file use redis-server /path/to/redis.conf

1:M 22 Aug 12:19:21.939 \* Running mode=standalone, port=6379.

1:M 22 Aug 12:19:21.939 # WARNING: The TCP backlog setting of 511 cannot be enforced because /proc/sys/net/core/somaxconn is set to the lower value of 128.

1:M 22 Aug 12:19:21.939 # Server initialized

1:M 22 Aug 12:19:21.939 # WARNING overcommit\_memory is set to 0! Background save may fail under low memory condition. To fix this issue add 'vm.overcommit\_memory = 1' to /etc/sysctl.conf and then reboot orrun the command 'sysctl vm.overcommit\_memory=1' for this to take effect.

1:M 22 Aug 12:19:21.939 # WARNING you have Transparent Huge Pages (THP) support enabled in your kernel. This will create latency and memory usage issues with Redis. To fix this issue run the command 'echo never > /sys/kernel/mm/transparent\_hugepage/enabled' as root, and add it to your /etc/rc.local in orderto retain the setting after a reboot. Redis must be restarted after THP is disabled.

1:M 22 Aug 12:19:21.939 \* Ready to accept connections

$

$

$ docker run -d --name redisHostPort -p 6379:6379 redis:latest

9cb079be043ba2891dd7a934f179bc7a6c7a83a54fcc2133b91b1c5aa34b158f

$ docker run -d --name redisDynamic -p 6379 redis:latest

7a63ef1dd2f04409cc04d3d2b7d3242e5050108207441c988640d473d58f1cc2

$ docker port redisDynamic 6379

0.0.0.0:32768

$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

7a63ef1dd2f0 redis:latest "docker-entrypoint.s…" 19 seconds ago Up 18 seconds 0.0.0.0:32768->6379/tcp redisDynamic

9cb079be043b redis:latest "docker-entrypoint.s…" 48 seconds ago Up 47 seconds 0.0.0.0:6379->6379/tcp redisHostPort

330a642cfb1e redis "docker-entrypoint.s…" 3 minutes ago Up 3 minutes 6379/tcp upbeat\_davinci

$

$ docker run -d --name redisMapped -v /opt/docker/data/redis:/data redis

de2a71d81a8f656fad148fc2b785b7e9941802939b4d5d93fbd79624f1871de9

$

$

$ docker run ubuntu ps

PID TTY TIME CMD

1 ? 00:00:00 ps

$ docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

redis latest 4760dc956b2d 5 months ago 107MB

ubuntu latest f975c5035748 5 months ago 112MB

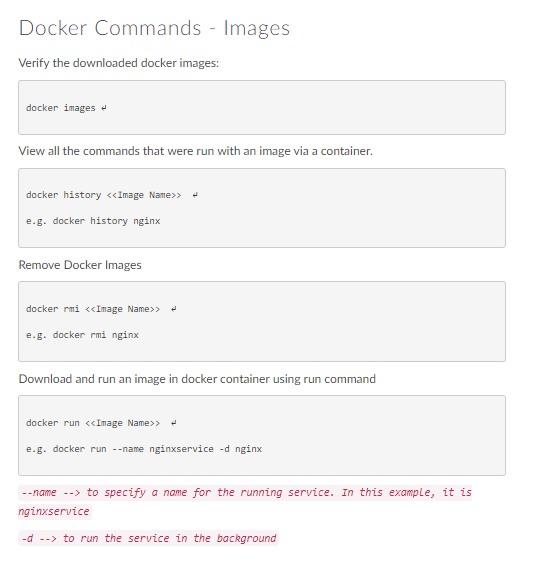
alpine latest 3fd9065eaf02 7 months ago 4.14MB

$

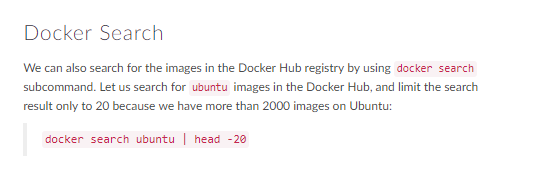
$ docker run -it ubuntu bash

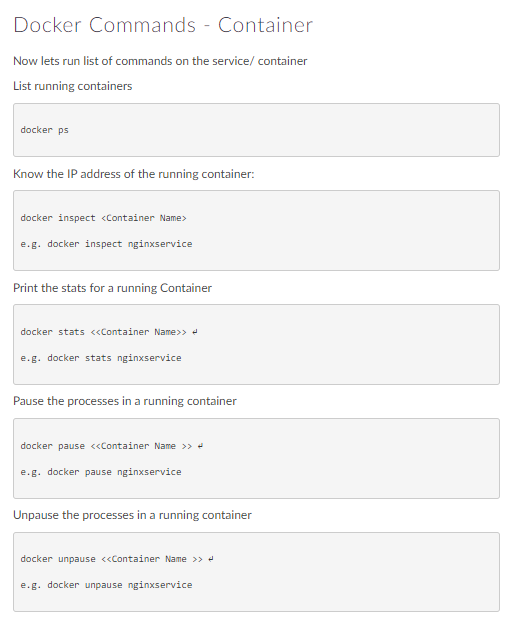
root@0315ded6a37a:/#exit





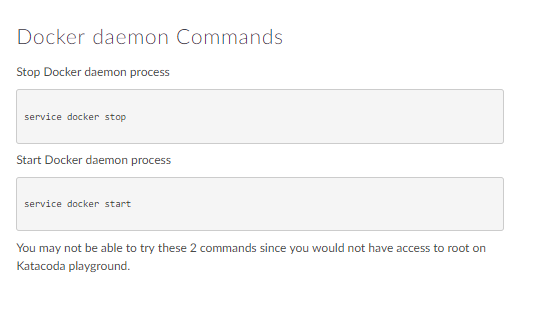


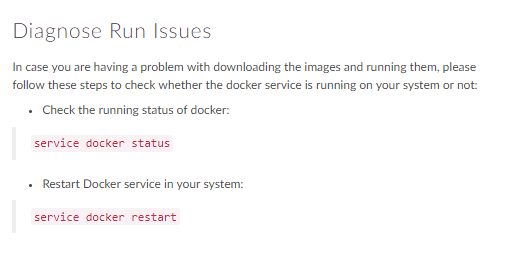


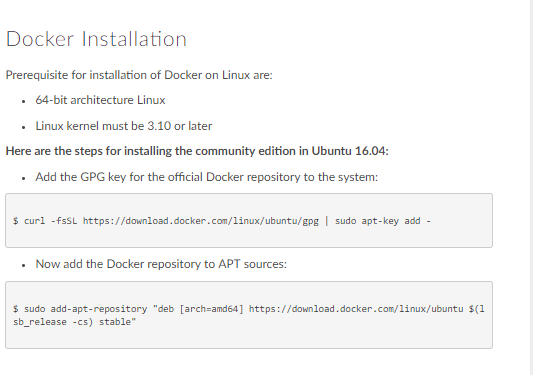


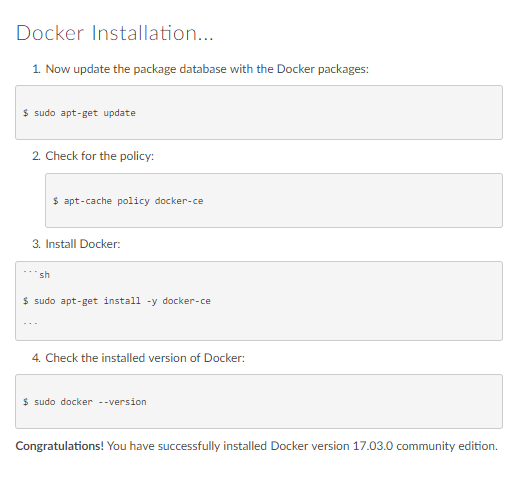


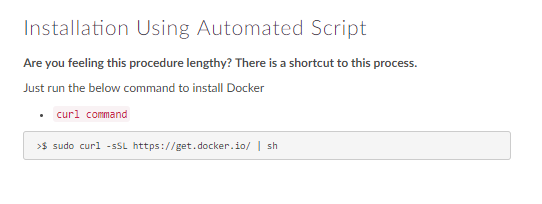


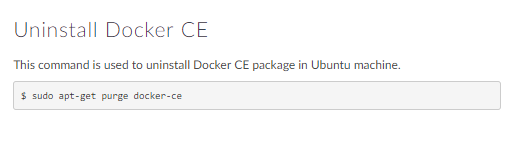




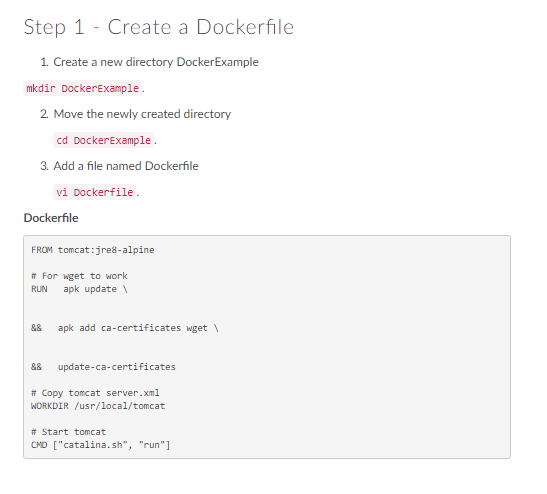


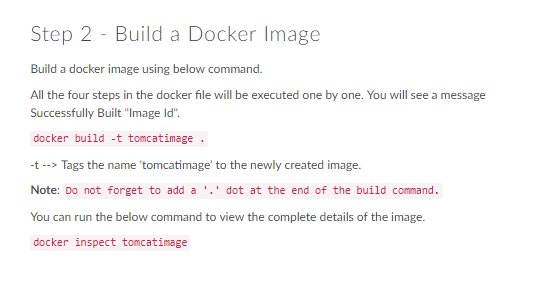


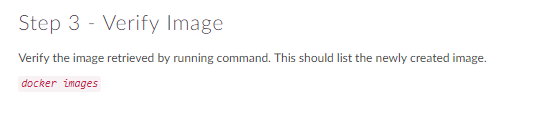


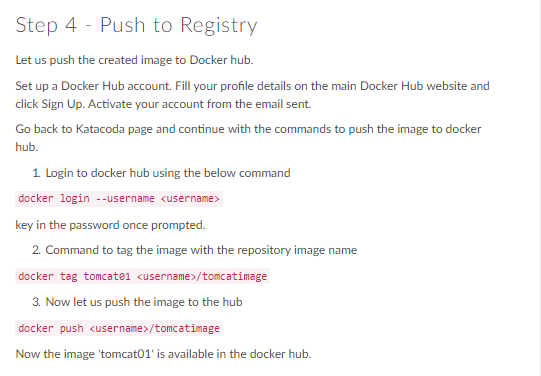


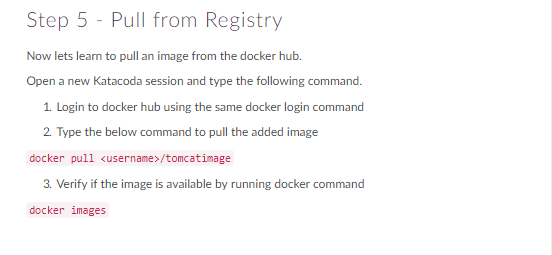
Docker example:







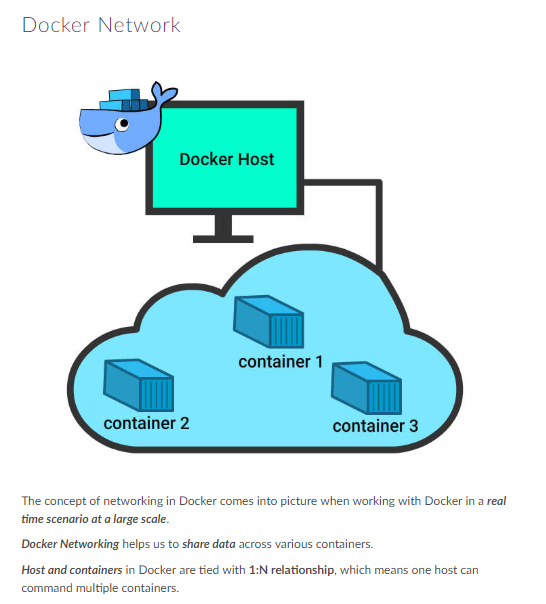


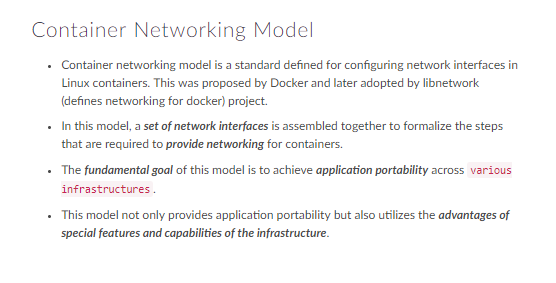


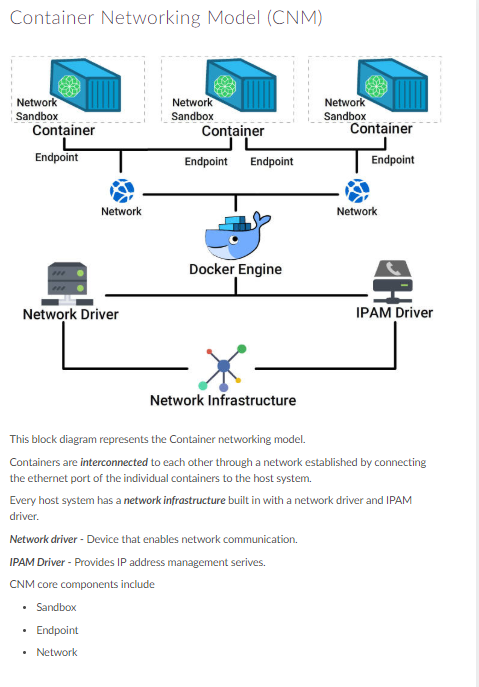


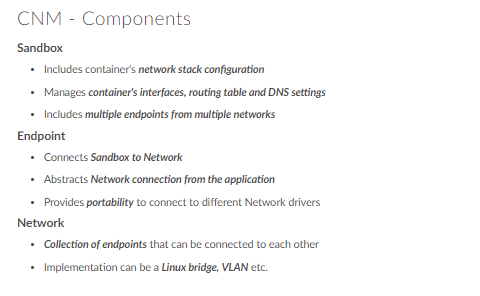


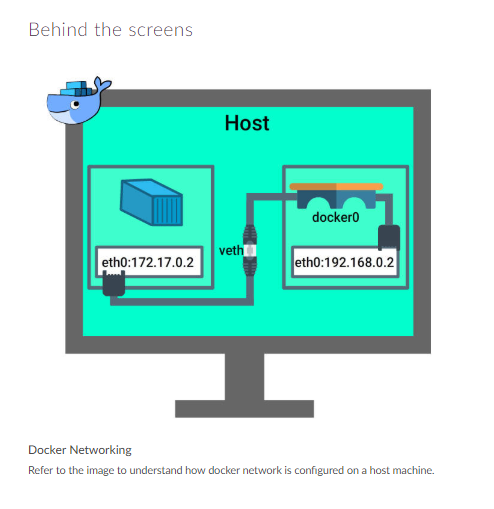
Docker Networking:

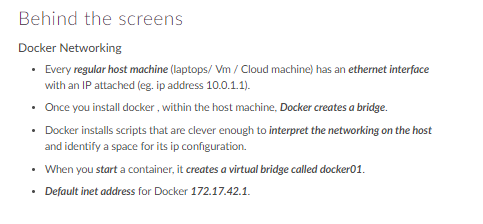


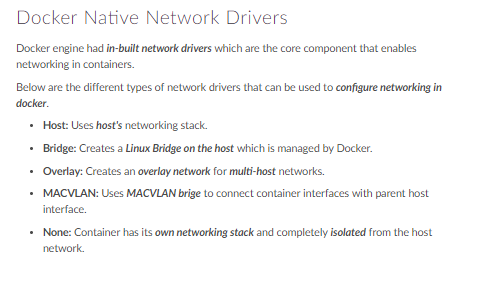


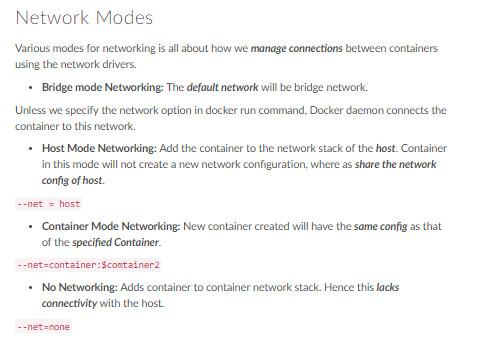


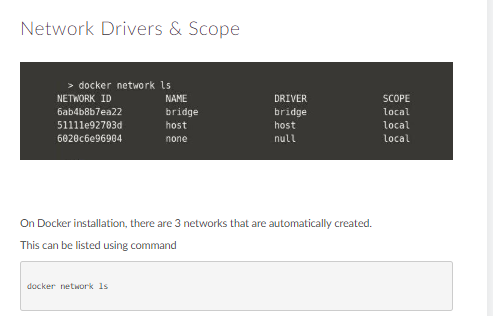


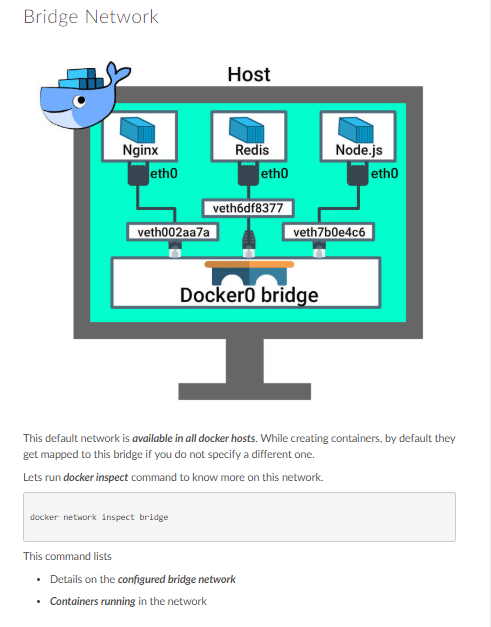


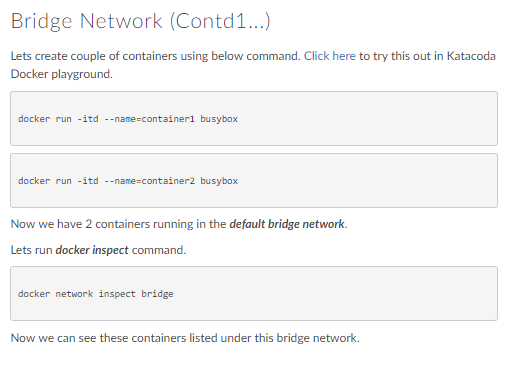


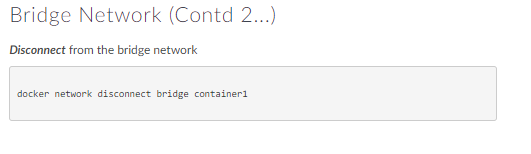


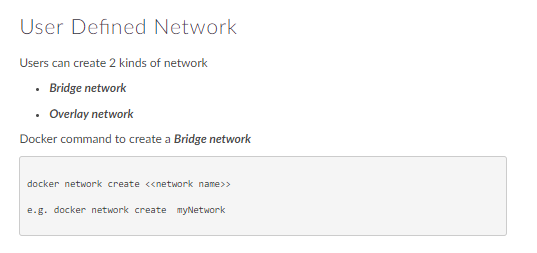


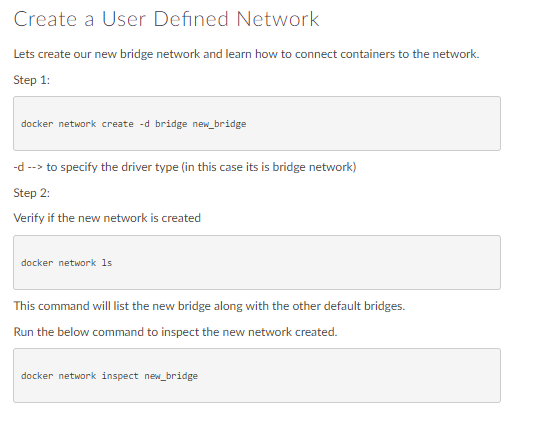


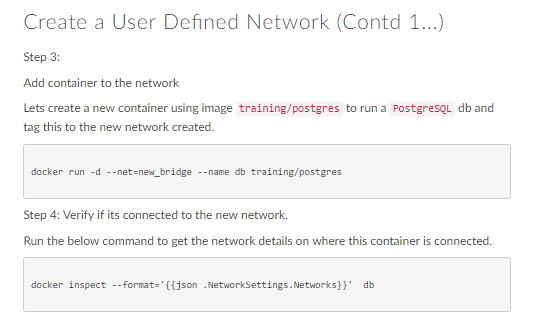


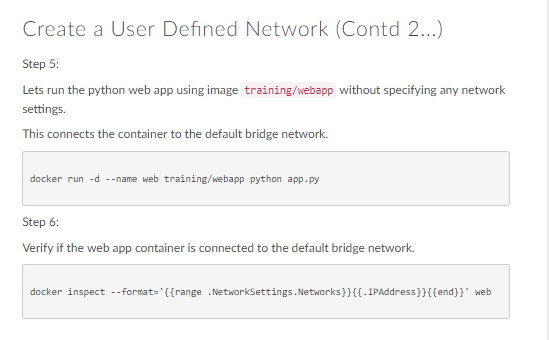


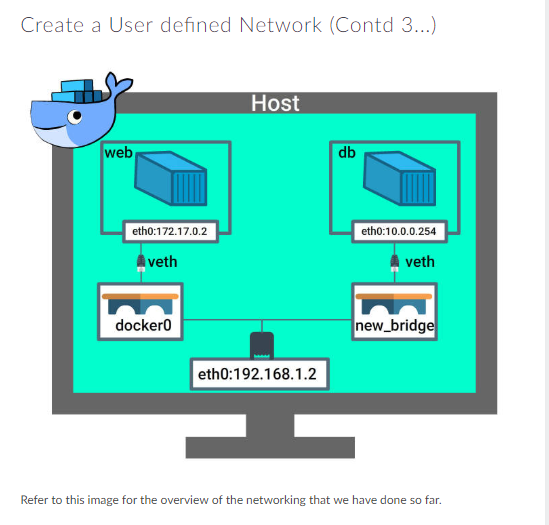


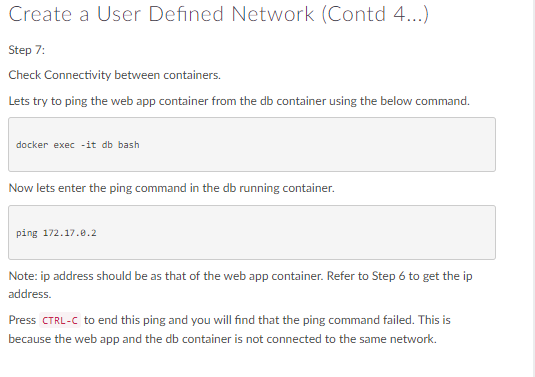


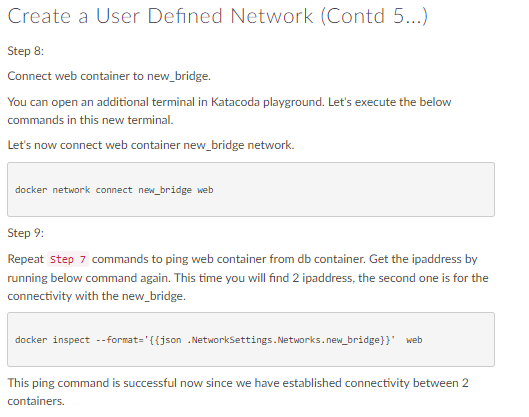




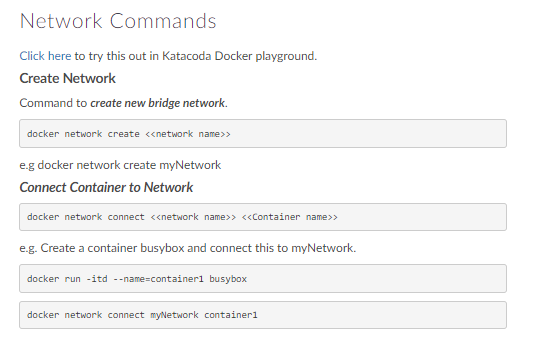


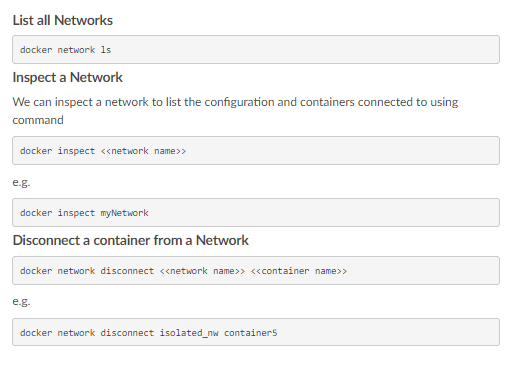


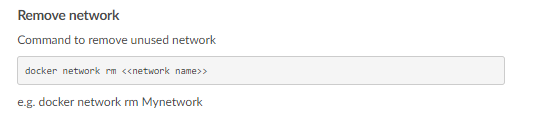












Hands-on:

Your Interactive Bash Terminal.

A good starting point is executing `docker`

$

$ docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

redis latest 4760dc956b2d 5 months ago 107MB

ubuntu latest f975c5035748 5 months ago 112MB

alpine latest 3fd9065eaf02 7 months ago 4.14MB

$

$ ls -lart

total 8

drwxr-xr-x 2 scrapbook scrapbook 4096 Mar 12 21:12 .

drwxr-xr-x 5 scrapbook scrapbook 4096 Jun 8 09:57 ..

$ vi dockerfile

$

$ ls -alrt

total 12

-rw-r--r-- 1 root root 79 Aug 23 11:37 dockerfile

drwxr-xr-x 1 scrapbook scrapbook 4096 Aug 23 11:37 ..

drwxr-xr-x 1 scrapbook scrapbook 4096 Aug 23 11:37 .

$

$ docker build -t ubuntuping:ubuntu .

Sending build context to Docker daemon 2.048kB

Step 1/3 : from ubuntu:latest

---> f975c5035748

Step 2/3 : run apt-get update && apt-get install iputils-ping

---> Running in 5f05546b40f9

Get:1 http://security.ubuntu.com/ubuntu xenial-security InRelease [107 kB]

Get:2 http://archive.ubuntu.com/ubuntu xenial InRelease [247 kB]

Get:3 http://archive.ubuntu.com/ubuntu xenial-updates InRelease [109 kB]

Get:4 http://archive.ubuntu.com/ubuntu xenial-backports InRelease [107 kB]

Get:5 http://security.ubuntu.com/ubuntu xenial-security/universe Sources [87.8 kB]

Get:6 http://security.ubuntu.com/ubuntu xenial-security/main amd64 Packages [691 kB]

Get:7 http://archive.ubuntu.com/ubuntu xenial/universe Sources [9802 kB]

Get:8 http://security.ubuntu.com/ubuntu xenial-security/restricted amd64 Packages [12.7 kB]

Get:9 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 Packages [463 kB]

Get:10 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 Packages [3748 B]

Get:11 http://archive.ubuntu.com/ubuntu xenial/main amd64 Packages [1558 kB]

Get:12 http://archive.ubuntu.com/ubuntu xenial/restricted amd64 Packages [14.1 kB]

Get:13 http://archive.ubuntu.com/ubuntu xenial/universe amd64 Packages [9827 kB]

Get:14 http://archive.ubuntu.com/ubuntu xenial/multiverse amd64 Packages [176 kB]

Get:15 http://archive.ubuntu.com/ubuntu xenial-updates/universe Sources [274 kB]

Get:16 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 Packages [1076 kB]

Get:17 http://archive.ubuntu.com/ubuntu xenial-updates/restricted amd64 Packages [13.1 kB]

Get:18 http://archive.ubuntu.com/ubuntu xenial-updates/universe amd64 Packages [873 kB]

Get:19 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 Packages [18.8 kB]

Get:20 http://archive.ubuntu.com/ubuntu xenial-backports/main amd64 Packages [7343 B]

Get:21 http://archive.ubuntu.com/ubuntu xenial-backports/universe amd64 Packages [8086 B]

Fetched 25.5 MB in 2s (10.4 MB/s)

Reading package lists...

Reading package lists...

Building dependency tree...

Reading state information...

The following additional packages will be installed:

libffi6 libgmp10 libgnutls-openssl27 libgnutls30 libhogweed4 libidn11

libnettle6 libp11-kit0 libtasn1-6

Suggested packages:

gnutls-bin

The following NEW packages will be installed:

iputils-ping libffi6 libgmp10 libgnutls-openssl27 libgnutls30 libhogweed4

libidn11 libnettle6 libp11-kit0 libtasn1-6

0 upgraded, 10 newly installed, 0 to remove and 29 not upgraded.

Need to get 1305 kB of archives.

After this operation, 3779 kB of additional disk space will be used.

Do you want to continue? [Y/n] Abort.

The command '/bin/sh -c apt-get update && apt-get install iputils-ping' returned a non-zero code: 1

$

$ vi dockerfile

$

$

$ docker build -t ubuntuwthping:ubuntu .

Sending build context to Docker daemon 2.048kB

Step 1/3 : from ubuntu:latest

---> f975c5035748

Step 2/3 : run apt-get update && apt-get install -y iputils-ping

---> Running in 2e13dbb3ad36

Get:1 http://archive.ubuntu.com/ubuntu xenial InRelease [247 kB]

Get:2 http://security.ubuntu.com/ubuntu xenial-security InRelease [107 kB]

Get:3 http://archive.ubuntu.com/ubuntu xenial-updates InRelease [109 kB]

Get:4 http://archive.ubuntu.com/ubuntu xenial-backports InRelease [107 kB]

Get:5 http://archive.ubuntu.com/ubuntu xenial/universe Sources [9802 kB]

Get:6 http://archive.ubuntu.com/ubuntu xenial/main amd64 Packages [1558 kB]

Get:7 http://archive.ubuntu.com/ubuntu xenial/restricted amd64 Packages [14.1 kB]

Get:8 http://archive.ubuntu.com/ubuntu xenial/universe amd64 Packages [9827 kB]

Get:9 http://archive.ubuntu.com/ubuntu xenial/multiverse amd64 Packages [176 kB]

Get:10 http://archive.ubuntu.com/ubuntu xenial-updates/universe Sources [274 kB]

Get:11 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 Packages [1076 kB]

Get:12 http://archive.ubuntu.com/ubuntu xenial-updates/restricted amd64 Packages [13.1 kB]

Get:13 http://archive.ubuntu.com/ubuntu xenial-updates/universe amd64 Packages [873 kB]

Get:14 http://security.ubuntu.com/ubuntu xenial-security/universe Sources [87.8 kB]

Get:15 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 Packages [18.8 kB]

Get:16 http://archive.ubuntu.com/ubuntu xenial-backports/main amd64 Packages [7343 B]

Get:17 http://archive.ubuntu.com/ubuntu xenial-backports/universe amd64 Packages [8086 B]

Get:18 http://security.ubuntu.com/ubuntu xenial-security/main amd64 Packages [691 kB]

Get:19 http://security.ubuntu.com/ubuntu xenial-security/restricted amd64 Packages [12.7 kB]

Get:20 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 Packages [463 kB]

Get:21 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 Packages [3748 B]

Fetched 25.5 MB in 2s (10.6 MB/s)

Reading package lists...

Reading package lists...

Building dependency tree...

Reading state information...

The following additional packages will be installed:

libffi6 libgmp10 libgnutls-openssl27 libgnutls30 libhogweed4 libidn11

libnettle6 libp11-kit0 libtasn1-6

Suggested packages:

gnutls-bin

The following NEW packages will be installed:

iputils-ping libffi6 libgmp10 libgnutls-openssl27 libgnutls30 libhogweed4

libidn11 libnettle6 libp11-kit0 libtasn1-6

0 upgraded, 10 newly installed, 0 to remove and 29 not upgraded.

Need to get 1305 kB of archives.

After this operation, 3779 kB of additional disk space will be used.

Get:1 http://archive.ubuntu.com/ubuntu xenial/main amd64 libgmp10 amd64 2:6.1.0+dfsg-2 [240 kB]

Get:2 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 libnettle6 amd64 3.2-1ubuntu0.16.04.1[93.5 kB]

Get:3 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 libhogweed4 amd64 3.2-1ubuntu0.16.04.1 [136 kB]

Get:4 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 libidn11 amd64 1.32-3ubuntu1.2 [46.5 kB]

Get:5 http://archive.ubuntu.com/ubuntu xenial/main amd64 libffi6 amd64 3.2.1-4 [17.8 kB]

Get:6 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 libp11-kit0 amd64 0.23.2-5~ubuntu16.04.1 [105 kB]

Get:7 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 libtasn1-6 amd64 4.7-3ubuntu0.16.04.3[43.5 kB]

Get:8 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 libgnutls30 amd64 3.4.10-4ubuntu1.4 [548 kB]

Get:9 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 libgnutls-openssl27 amd64 3.4.10-4ubuntu1.4 [22.0 kB]

Get:10 http://archive.ubuntu.com/ubuntu xenial/main amd64 iputils-ping amd64 3:20121221-5ubuntu2 [52.7kB]

debconf: delaying package configuration, since apt-utils is not installed

Fetched 1305 kB in 0s (4987 kB/s)

Selecting previously unselected package libgmp10:amd64.

(Reading database ... 4768 files and directories currently installed.)

Preparing to unpack .../libgmp10\_2%3a6.1.0+dfsg-2\_amd64.deb ...

Unpacking libgmp10:amd64 (2:6.1.0+dfsg-2) ...

Selecting previously unselected package libnettle6:amd64.

Preparing to unpack .../libnettle6\_3.2-1ubuntu0.16.04.1\_amd64.deb ...

Unpacking libnettle6:amd64 (3.2-1ubuntu0.16.04.1) ...

Selecting previously unselected package libhogweed4:amd64.

Preparing to unpack .../libhogweed4\_3.2-1ubuntu0.16.04.1\_amd64.deb ...

Unpacking libhogweed4:amd64 (3.2-1ubuntu0.16.04.1) ...

Selecting previously unselected package libidn11:amd64.

Preparing to unpack .../libidn11\_1.32-3ubuntu1.2\_amd64.deb ...

Unpacking libidn11:amd64 (1.32-3ubuntu1.2) ...

Selecting previously unselected package libffi6:amd64.

Preparing to unpack .../libffi6\_3.2.1-4\_amd64.deb ...

Unpacking libffi6:amd64 (3.2.1-4) ...

Selecting previously unselected package libp11-kit0:amd64.

Preparing to unpack .../libp11-kit0\_0.23.2-5~ubuntu16.04.1\_amd64.deb ...

Unpacking libp11-kit0:amd64 (0.23.2-5~ubuntu16.04.1) ...

Selecting previously unselected package libtasn1-6:amd64.

Preparing to unpack .../libtasn1-6\_4.7-3ubuntu0.16.04.3\_amd64.deb ...

Unpacking libtasn1-6:amd64 (4.7-3ubuntu0.16.04.3) ...

Selecting previously unselected package libgnutls30:amd64.

Preparing to unpack .../libgnutls30\_3.4.10-4ubuntu1.4\_amd64.deb ...

Unpacking libgnutls30:amd64 (3.4.10-4ubuntu1.4) ...

Selecting previously unselected package libgnutls-openssl27:amd64.

Preparing to unpack .../libgnutls-openssl27\_3.4.10-4ubuntu1.4\_amd64.deb ...

Unpacking libgnutls-openssl27:amd64 (3.4.10-4ubuntu1.4) ...

Selecting previously unselected package iputils-ping.

Preparing to unpack .../iputils-ping\_3%3a20121221-5ubuntu2\_amd64.deb ...

Unpacking iputils-ping (3:20121221-5ubuntu2) ...

Processing triggers for libc-bin (2.23-0ubuntu10) ...

Setting up libgmp10:amd64 (2:6.1.0+dfsg-2) ...

Setting up libnettle6:amd64 (3.2-1ubuntu0.16.04.1) ...

Setting up libhogweed4:amd64 (3.2-1ubuntu0.16.04.1) ...

Setting up libidn11:amd64 (1.32-3ubuntu1.2) ...

Setting up libffi6:amd64 (3.2.1-4) ...

Setting up libp11-kit0:amd64 (0.23.2-5~ubuntu16.04.1) ...

Setting up libtasn1-6:amd64 (4.7-3ubuntu0.16.04.3) ...

Setting up libgnutls30:amd64 (3.4.10-4ubuntu1.4) ...

Setting up libgnutls-openssl27:amd64 (3.4.10-4ubuntu1.4) ...

Setting up iputils-ping (3:20121221-5ubuntu2) ...

Setcap is not installed, falling back to setuid

Processing triggers for libc-bin (2.23-0ubuntu10) ...

Removing intermediate container 2e13dbb3ad36

---> 5e1ba66b88b0

Step 3/3 : cmd bash

---> Running in 553eaebe17e7

Removing intermediate container 553eaebe17e7

---> 2ad9824038ed

Successfully built 2ad9824038ed

Successfully tagged ubuntuwthping:ubuntu

$

$

$

$ docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

ubuntuwthping ubuntu 2ad9824038ed 9 seconds ago 157MB

redis latest 4760dc956b2d 5 months ago 107MB

ubuntu latest f975c5035748 5 months ago 112MB

alpine latest 3fd9065eaf02 7 months ago 4.14MB

$

$

$ docker run -tid --name cont1 ubuntuwthping

Unable to find image 'ubuntuwthping:latest' locally

docker: Error response from daemon: pull access denied for ubuntuwthping, repository does not exist ormay require 'docker login'.

See 'docker run --help'.

$

$

$ docker run -tid --name cont1 ubuntuwthping:ubuntu

048ab40f8cd83c1f7aaaedebe8857cf7a72afcb4640dad217f0d8f027d2772bd

$

$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUSPORTS NAMES

048ab40f8cd8 ubuntuwthping:ubuntu "/bin/sh -c bash" 15 seconds ago Up 14 seconds cont1

$

$ docker network ls

NETWORK ID NAME DRIVER SCOPE

99a31b24f196 bridge bridge local

fa054a9af353 host host local

f50397115ef2 none null local

$

$ docker network inspect bridge

[

{

"Name": "bridge",

"Id": "99a31b24f196000ece4a728185930943cc8f638ac95e1225344d5471711b9438",

"Created": "2018-08-23T11:32:39.651101721Z",

"Scope": "local",

"Driver": "bridge",

"EnableIPv6": false,

"IPAM": {

"Driver": "default",

"Options": null,

"Config": [

{

"Subnet": "172.18.0.1/24",

"Gateway": "172.18.0.1"

}

]

},

"Internal": false,

"Attachable": false,

"Ingress": false,

"ConfigFrom": {

"Network": ""

},

"ConfigOnly": false,

"Containers": {

"048ab40f8cd83c1f7aaaedebe8857cf7a72afcb4640dad217f0d8f027d2772bd": {

"Name": "cont1",

"EndpointID": "6e529394f5c2e28f40cffc37fa7ad37e7b599b5bd25ad5bbf59402d7d5694887",

"MacAddress": "02:42:ac:12:00:02",

"IPv4Address": "172.18.0.2/24",

"IPv6Address": ""

}

},

"Options": {

"com.docker.network.bridge.default\_bridge": "true",

"com.docker.network.bridge.enable\_icc": "true",

"com.docker.network.bridge.enable\_ip\_masquerade": "true",

"com.docker.network.bridge.host\_binding\_ipv4": "0.0.0.0",

"com.docker.network.bridge.name": "docker0",

"com.docker.network.driver.mtu": "1500"

},

"Labels": {}

}

]

$

$ docker network create new\_bridge

1c67db8388b45f7094a8729506bb1709fe3fa8c901e61900b480b79ddd482767

$

$ docker run -tid --name cont2 --net=new\_bridge ubuntuwthping:ubuntu

781583bcdcc4e7cd1b85df404ddc31e346d44babd6dd2a74eb4809f18ab3dbdc

$

$ docker network new\_bridge connect cont1

Usage: docker network COMMAND

Manage networks

Commands:

connect Connect a container to a network

create Create a network

disconnect Disconnect a container from a network

inspect Display detailed information on one or more networks

ls List networks

prune Remove all unused networks

rm Remove one or more networks

Run 'docker network COMMAND --help' for more information on a command.

$ docker network connect new\_bridge cont1

$

$ docker network inspect new\_bridge

[

{

"Name": "new\_bridge",

"Id": "1c67db8388b45f7094a8729506bb1709fe3fa8c901e61900b480b79ddd482767",

"Created": "2018-08-23T11:42:52.550549548Z",

"Scope": "local",

"Driver": "bridge",

"EnableIPv6": false,

"IPAM": {

"Driver": "default",

"Options": {},

"Config": [

{

"Subnet": "172.19.0.0/16",

"Gateway": "172.19.0.1"

}

]

},

"Internal": false,

"Attachable": false,

"Ingress": false,

"ConfigFrom": {

"Network": ""

},

"ConfigOnly": false,

"Containers": {

"048ab40f8cd83c1f7aaaedebe8857cf7a72afcb4640dad217f0d8f027d2772bd": {

"Name": "cont1",

"EndpointID": "069c4dbdf36d1e99b0e0df13998879c5c99cc1806a22766b14c0ae274579c278",

"MacAddress": "02:42:ac:13:00:03",

"IPv4Address": "172.19.0.3/16",

"IPv6Address": ""

},

"781583bcdcc4e7cd1b85df404ddc31e346d44babd6dd2a74eb4809f18ab3dbdc": {

"Name": "cont2",

"EndpointID": "202f5bf988f01c0dc61f8c14ef9d570b8e8d1a92f7e3bccece035b883ad2bcd0",

"MacAddress": "02:42:ac:13:00:02",

"IPv4Address": "172.19.0.2/16",

"IPv6Address": ""

}

},

"Options": {},

"Labels": {}

}

]

$ docker exec -ti cont1 bash

root@048ab40f8cd8:/# ping 172.19.0.2

PING 172.19.0.2 (172.19.0.2) 56(84) bytes of data.

64 bytes from 172.19.0.2: icmp\_seq=1 ttl=64 time=0.094 ms

64 bytes from 172.19.0.2: icmp\_seq=2 ttl=64 time=0.068 ms

64 bytes from 172.19.0.2: icmp\_seq=3 ttl=64 time=0.061 ms

^C

--- 172.19.0.2 ping statistics ---

3 packets transmitted, 3 received, 0% packet loss, time 2110ms

rtt min/avg/max/mdev = 0.061/0.074/0.094/0.015 ms

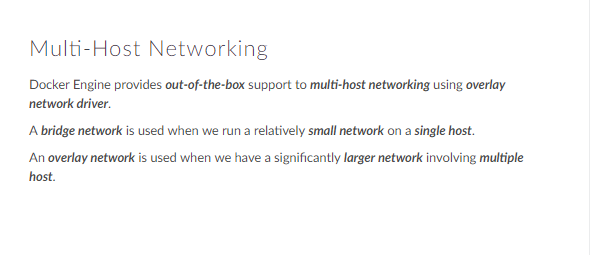
root@048ab40f8cd8:/# exit

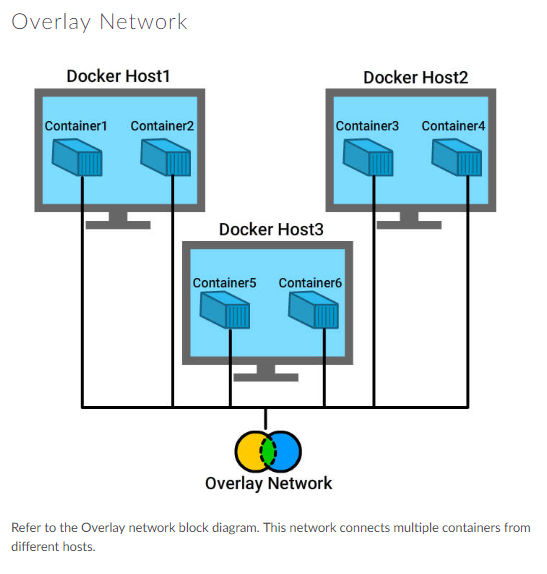
exit

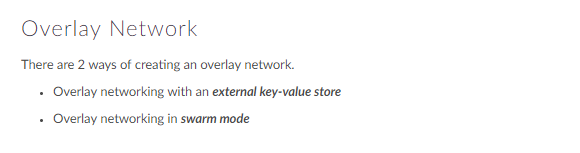
$ docker network disconnect new\_bridge cont1

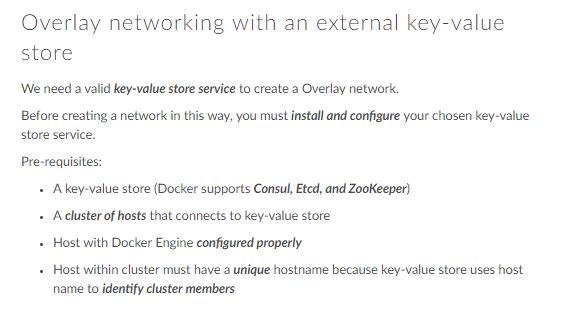
$ ^C

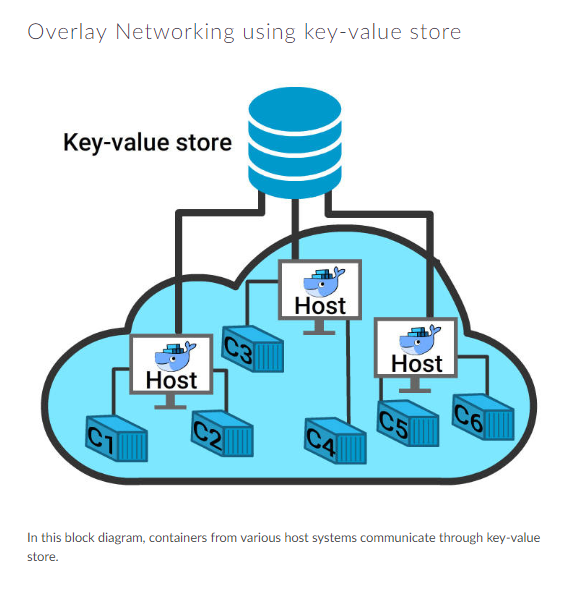
$



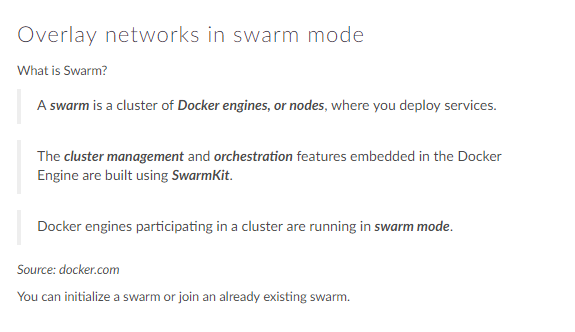


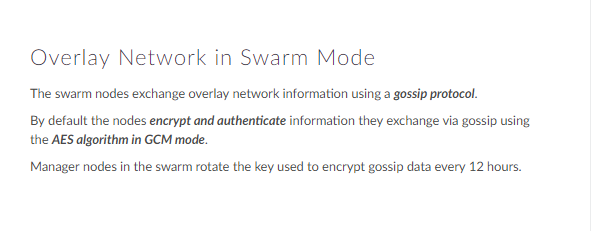






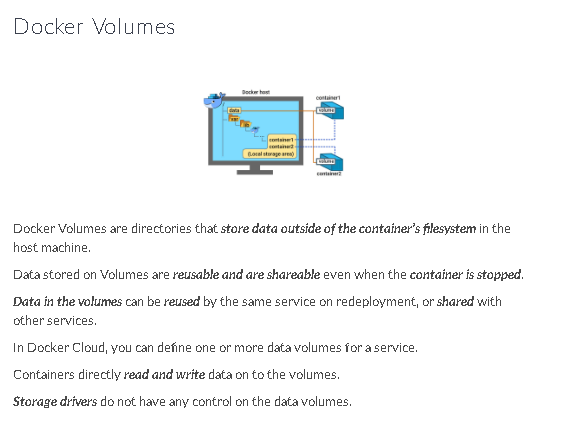
Key-value store: A *key-value database* (also known as a key-value store and key-value store database) is a type of [NoSQL](https://database.guide/what-is-nosql/) database that uses a simple key/value method to store data. A key-value is also commonly referred to as a dictionary or hash. examples are ip routing table and phone directory.

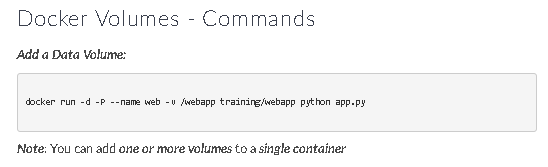




What is Gossip Protocol

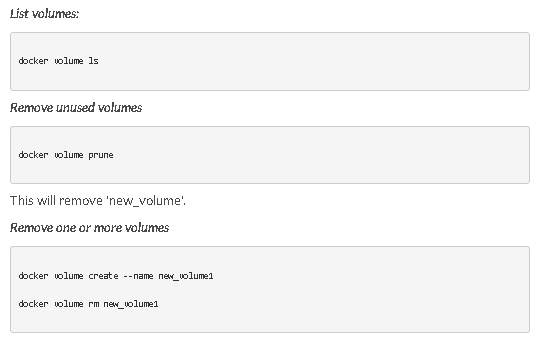
A **gossip protocol** is a style of computer-to-computer communication **protocol** inspired by the form of **gossip** seen in social networks. Modern distributed systems often use **gossip protocol**s to solve problems that might be difficult to solve in other ways, either because the underlying network has an inconvenient structure, is extremely large, or because **gossip** solutions are the most efficient ones available. **Gossip protocol**s are probabilistic in nature: a node chooses its partner node with which to communicate randomly. They are scalable because each node sends only a fixed number of messages, independent of the number of nodes in the network. In addition, a node does not wait for acknowledgments nor does it take some recovery action should an acknowledgment not arrive. They achieve fault-tolerance because a node receives copies of a message from different nodes. No node has a specific role to play, and so a failed node will not prevent other nodes from continuing sending messages. Hence, there is no need for failure detection or specific recovery actions.

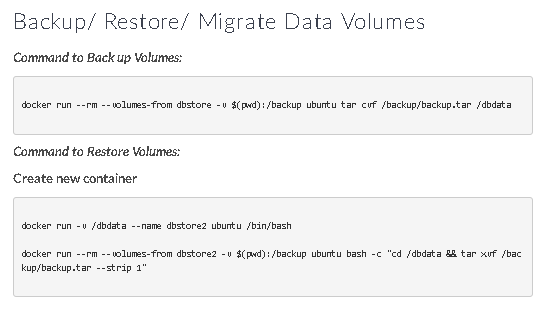


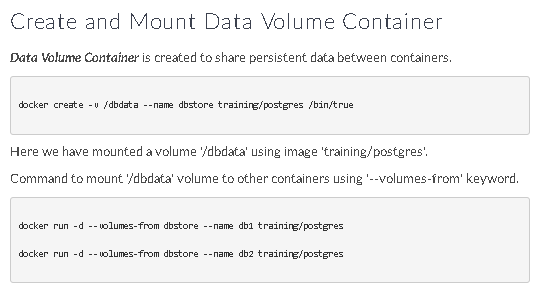


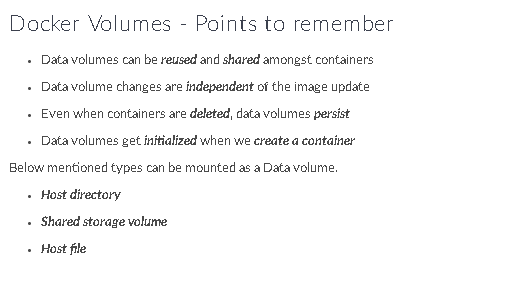


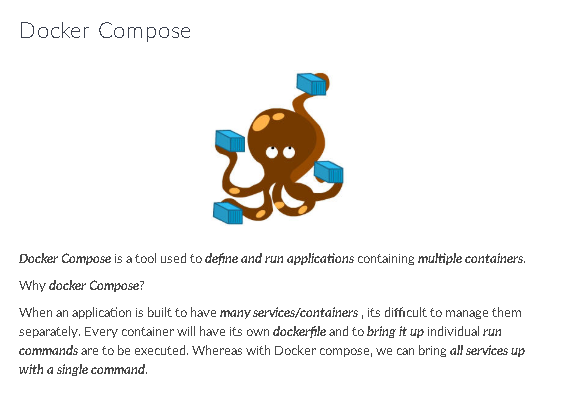




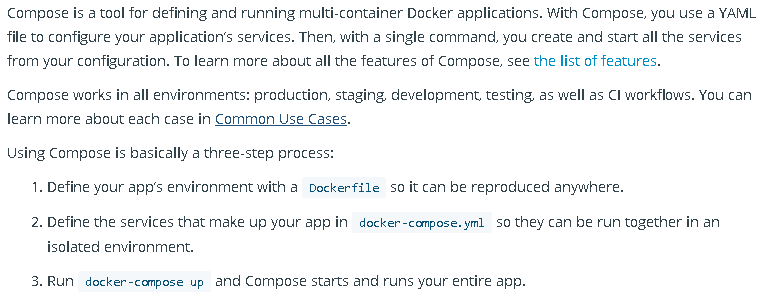




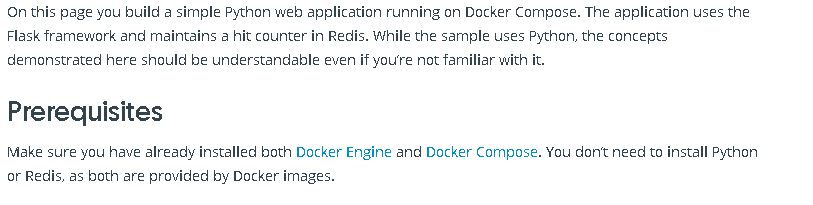




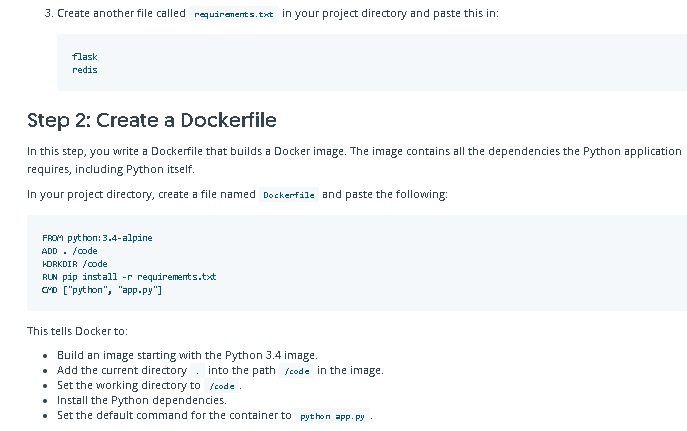


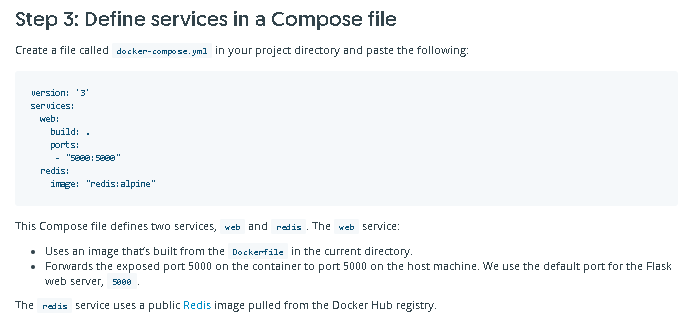


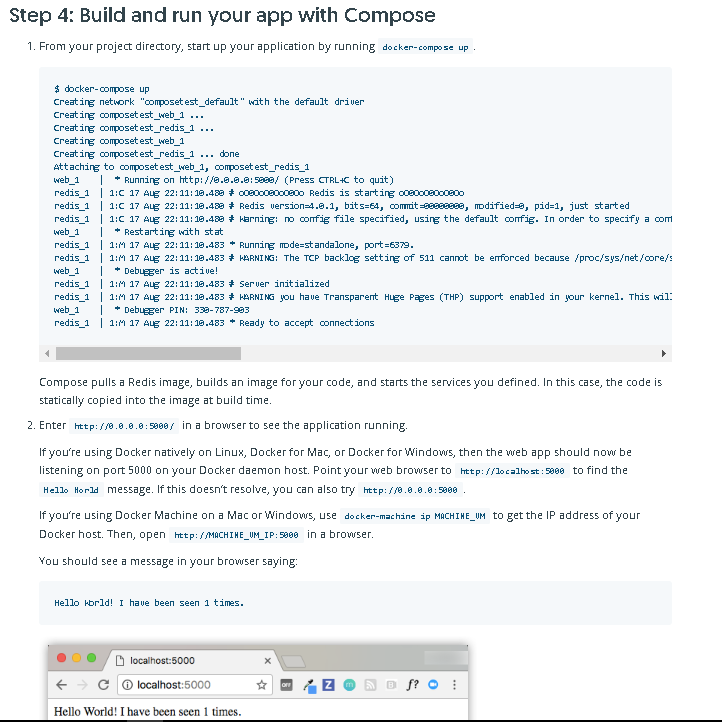
**Simple python web application using compose**

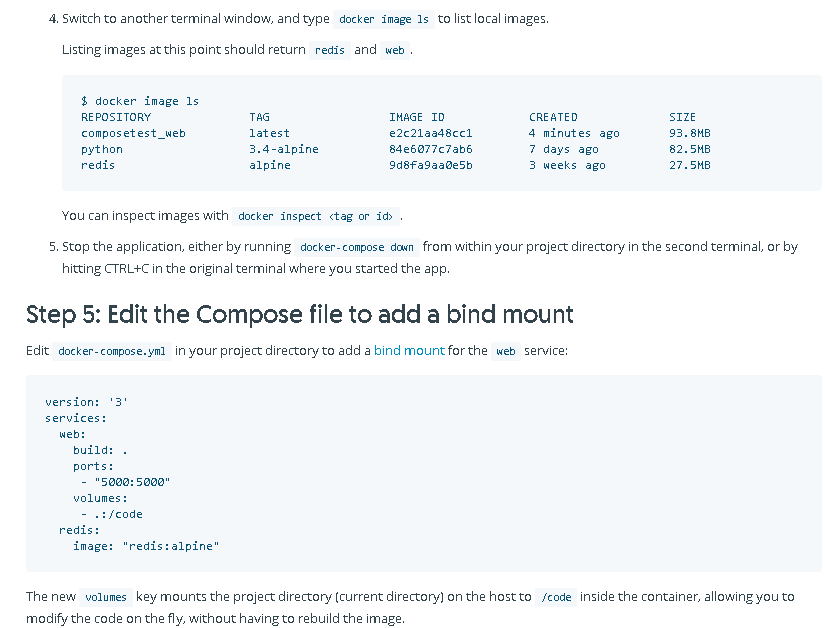


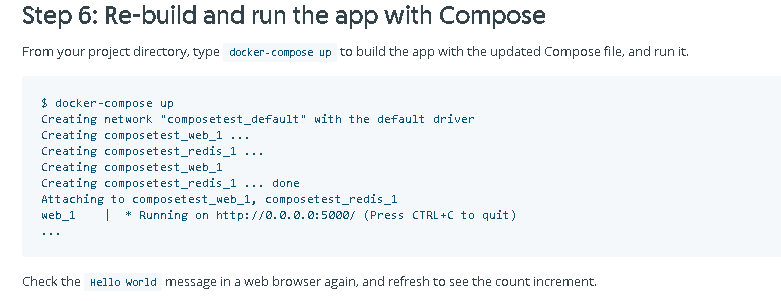


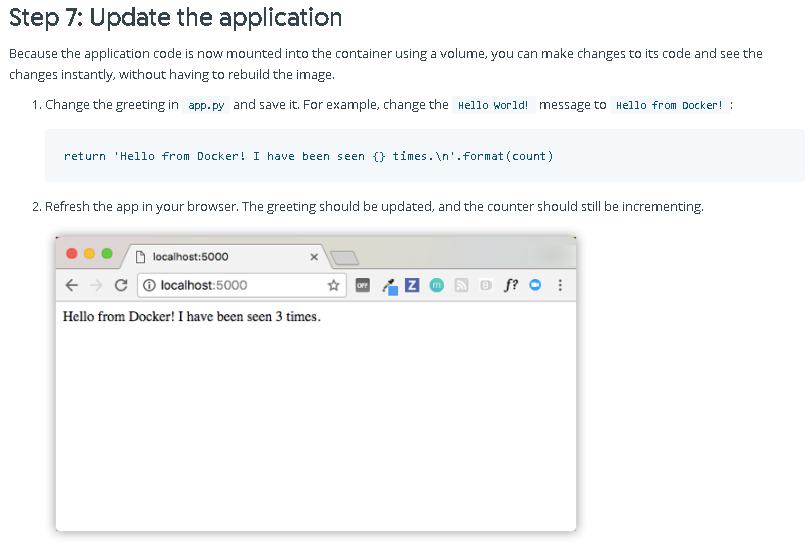












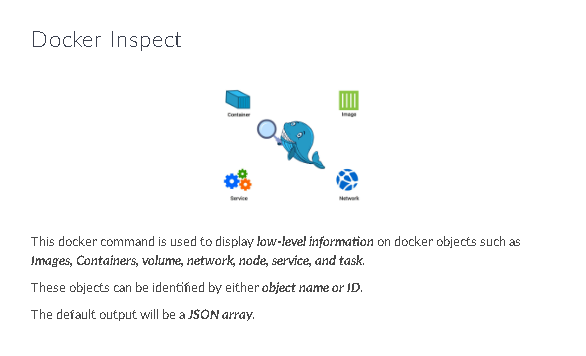
**Some usefull compose cmds:**

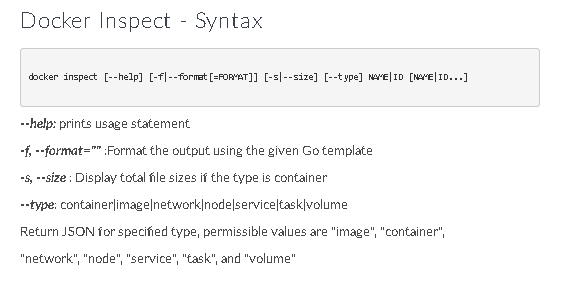
Docker-compose up –d **---for launching all services applications**

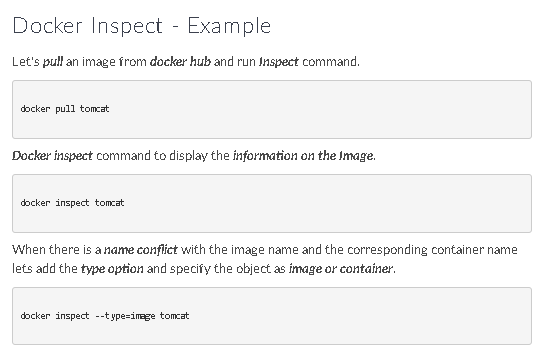
Docker-compose ps **---to see all services/containers**

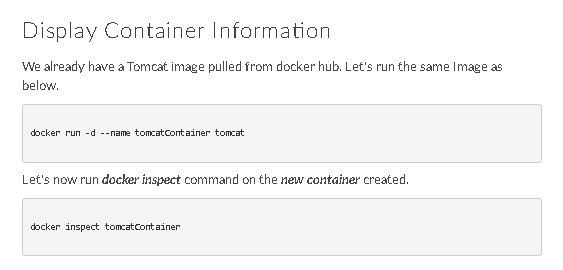
Docker-compose logs **---to see logs of all services/containers**

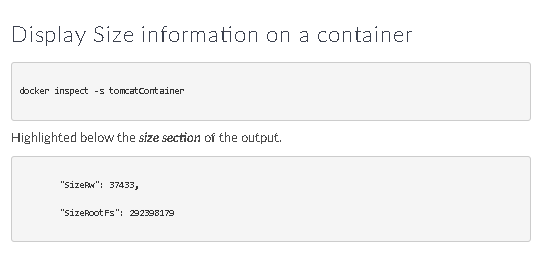
Docker-compose scale web(services name)=3 (no of instances) **---- to scale up or down the services/containers**

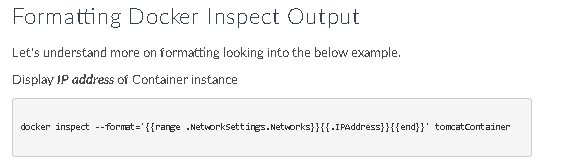




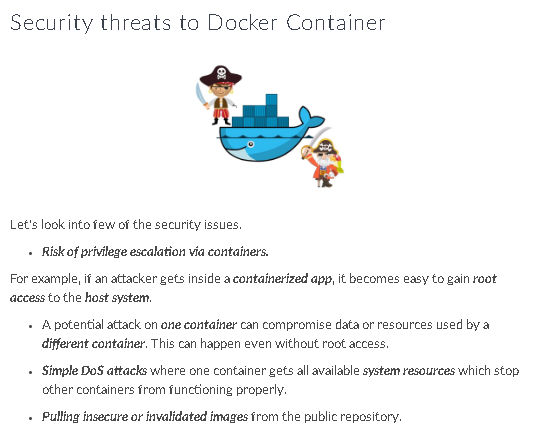


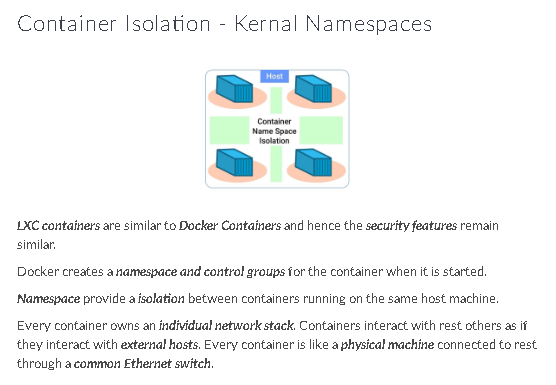


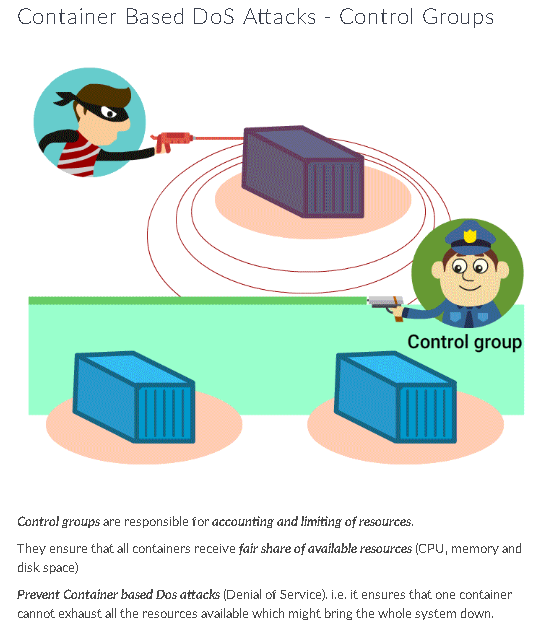


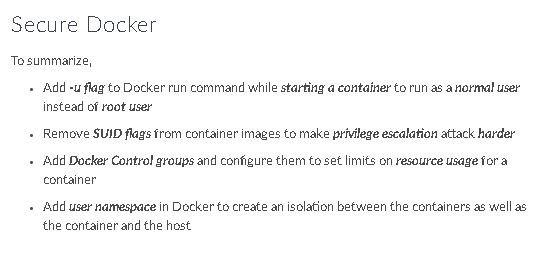




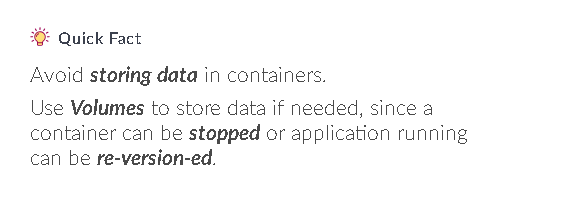




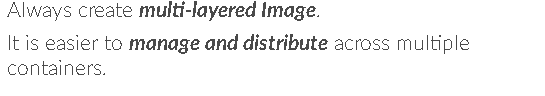


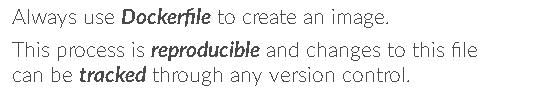


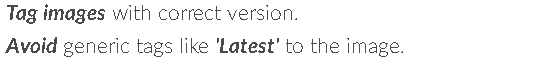
**Docker best practices:**

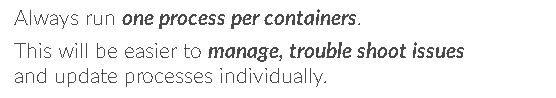


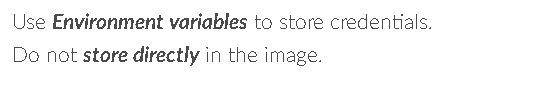




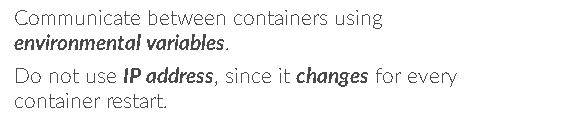


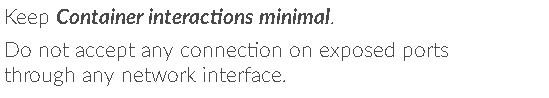






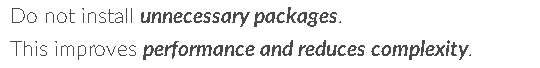


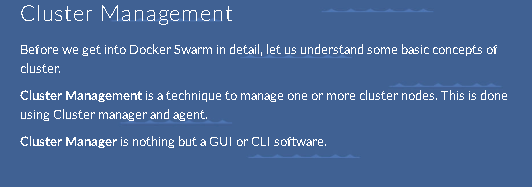


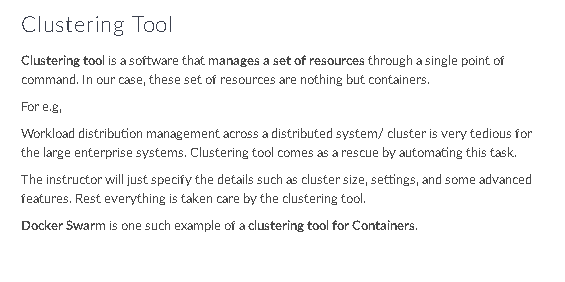


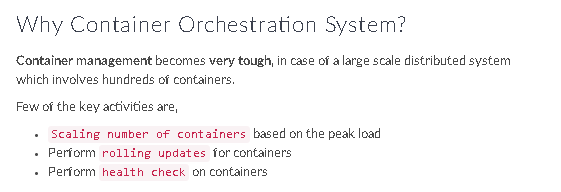


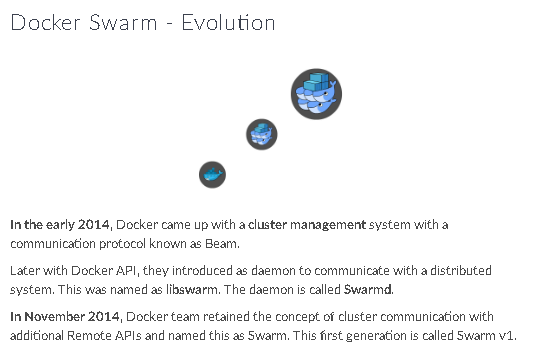


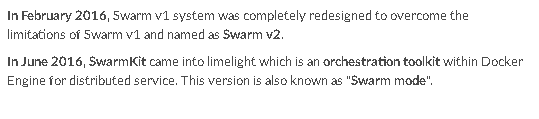


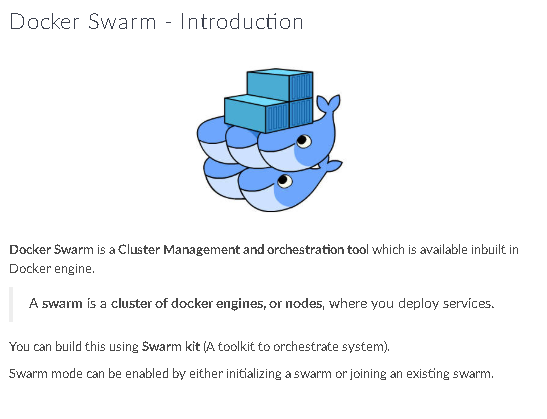


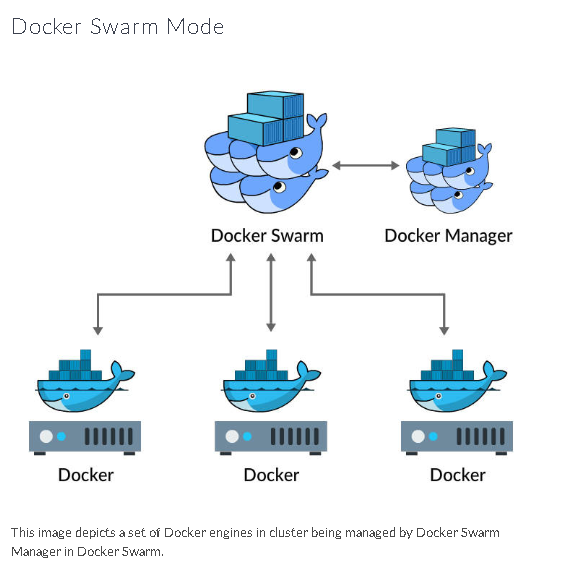


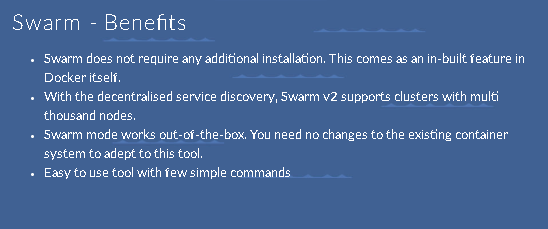


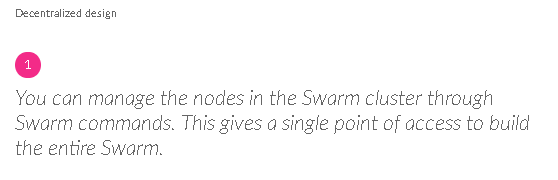


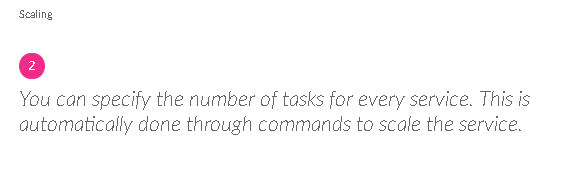


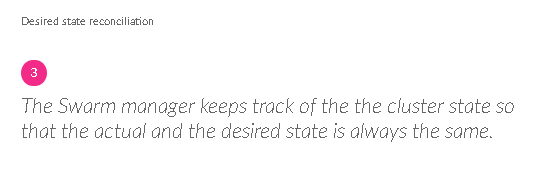


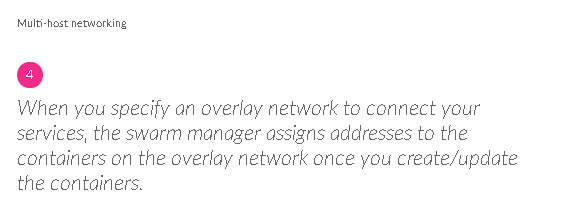


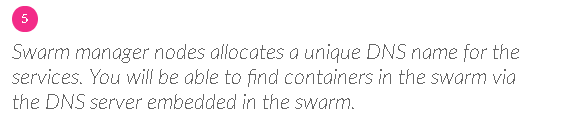


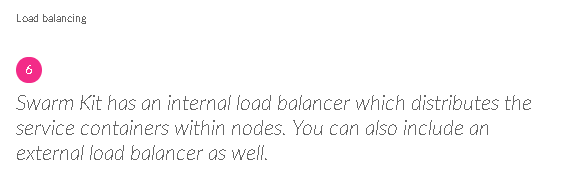




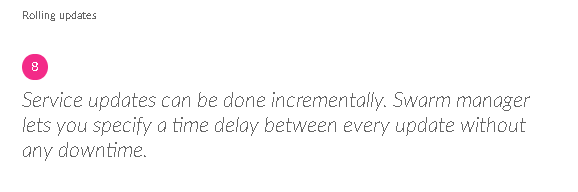




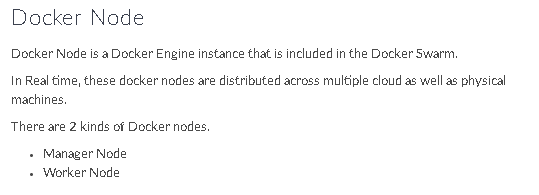


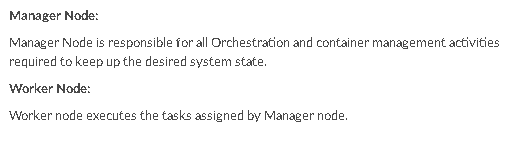


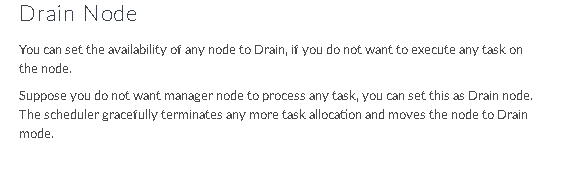


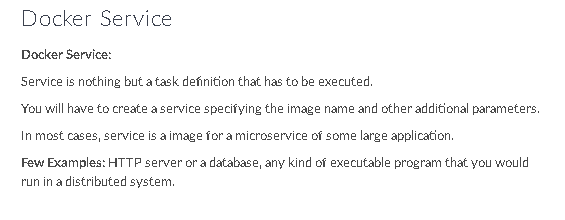


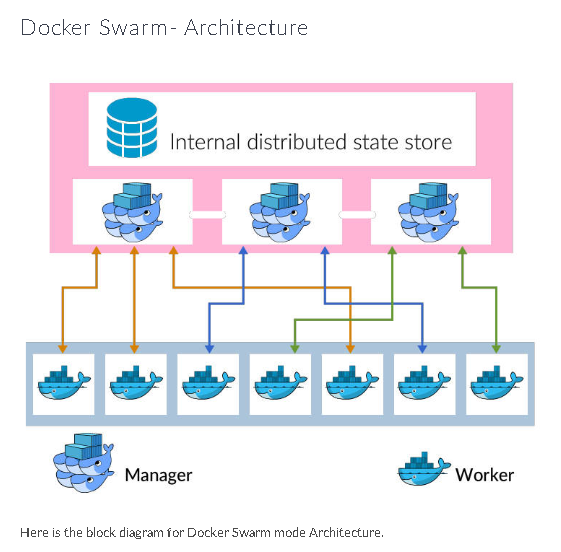


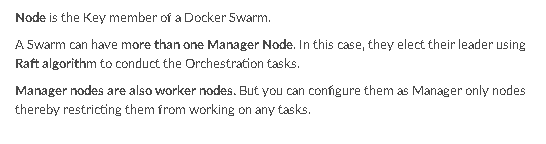


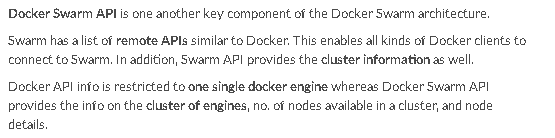


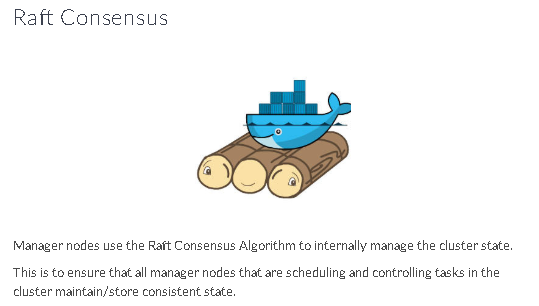




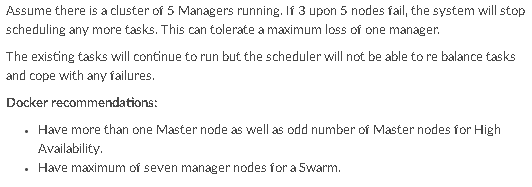


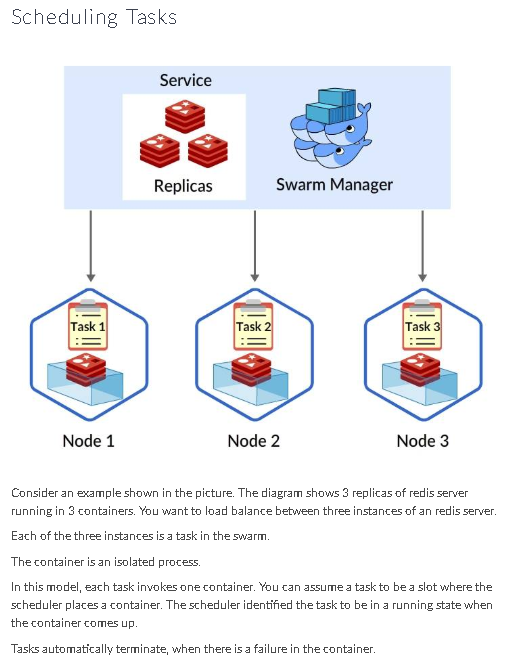


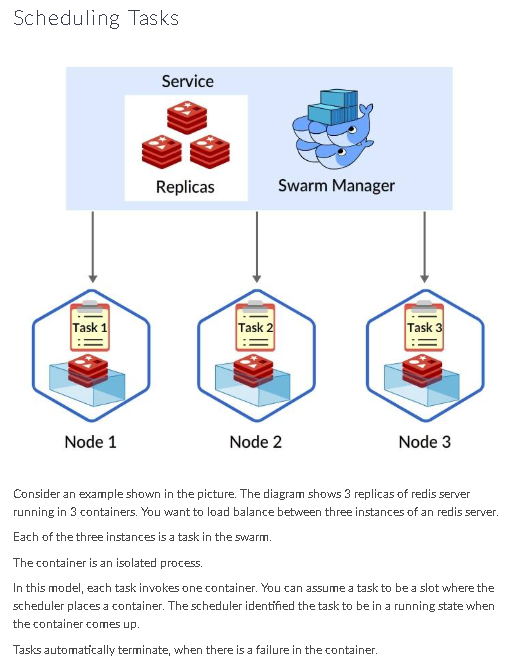


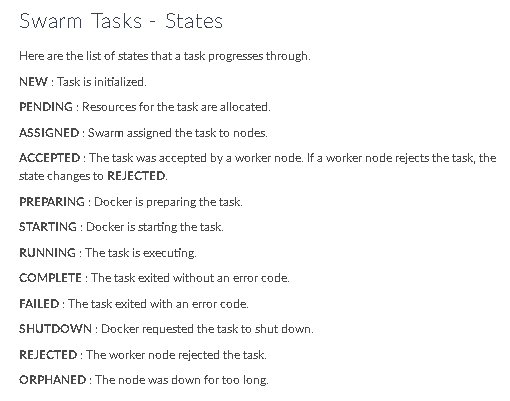


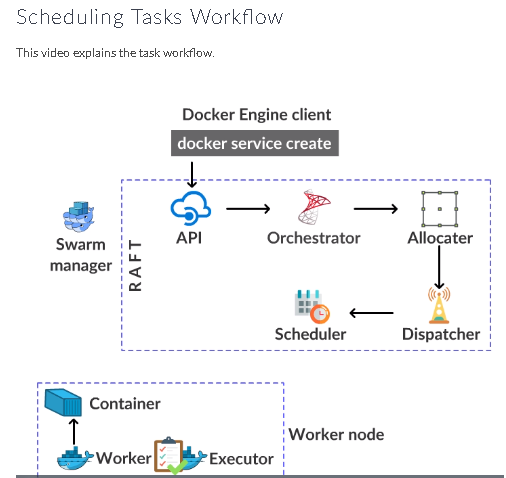


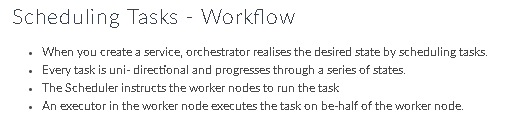


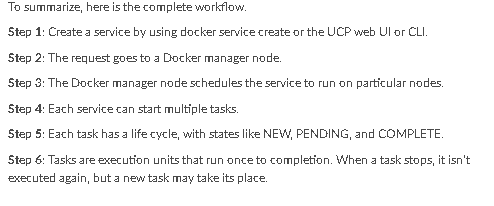


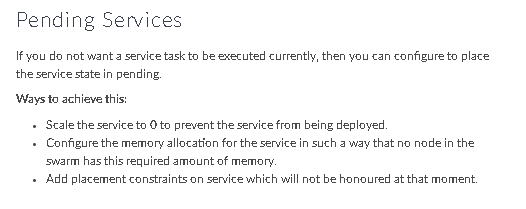


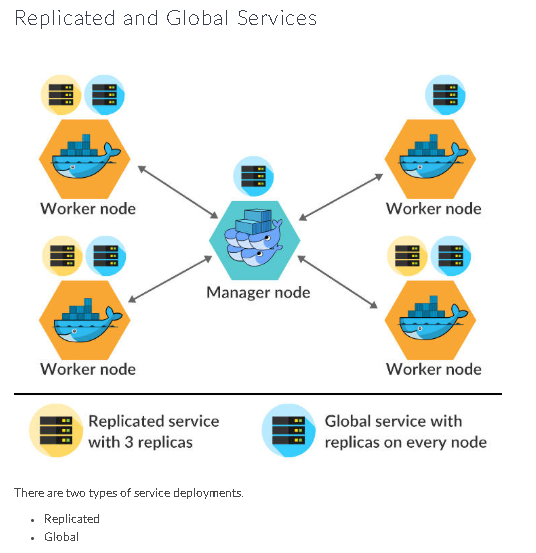


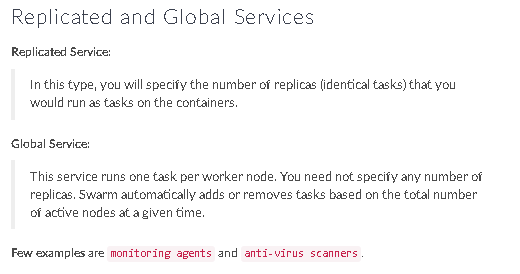


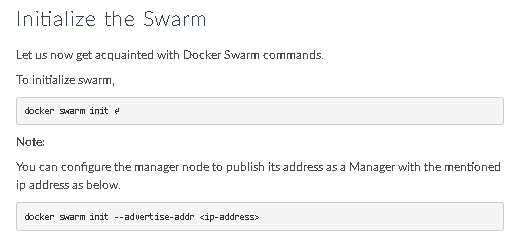


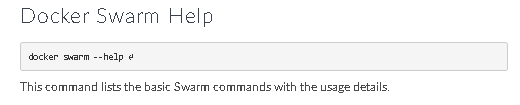


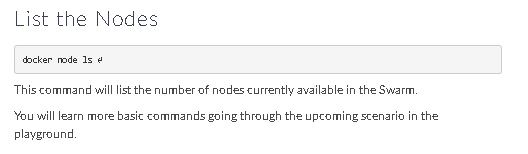


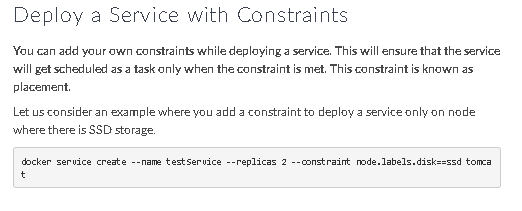




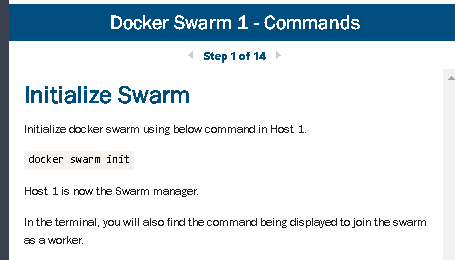


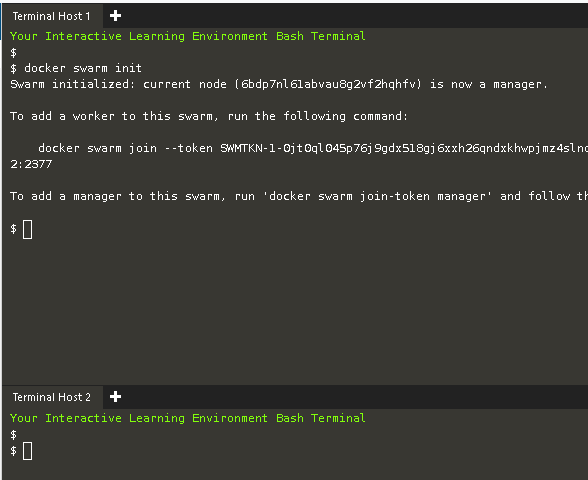


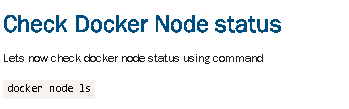


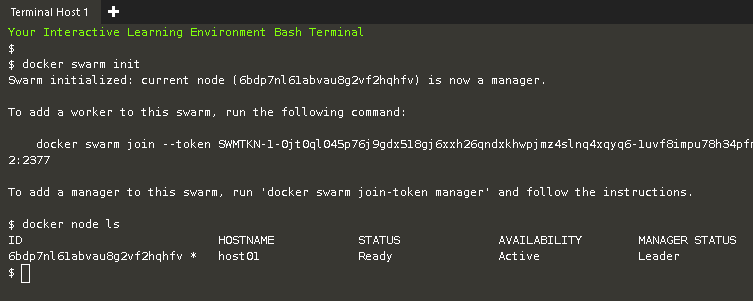


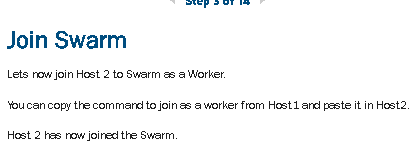
**Example:**

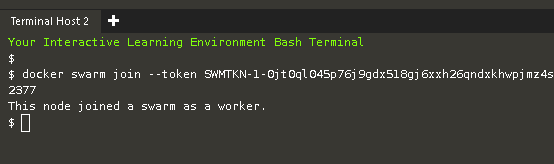


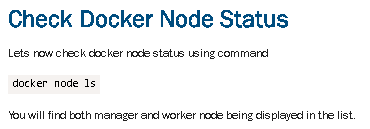


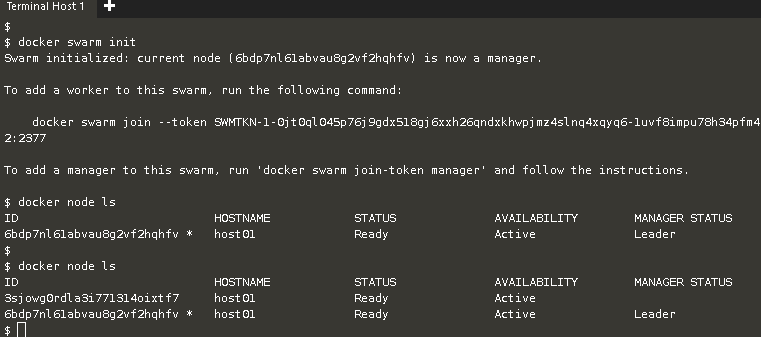








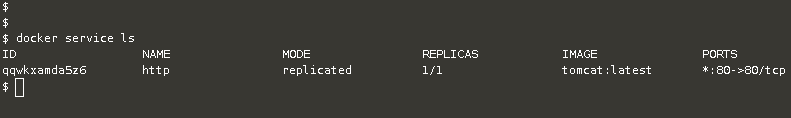




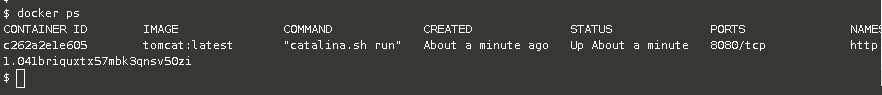


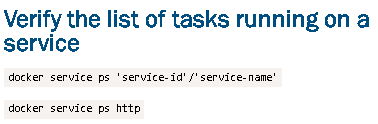




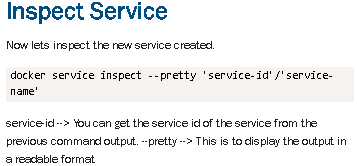












**$ docker service inspect --pretty http**

**ID: qqwkxamda5z67bwwflfyvaus8**

**Name: http**

**Service Mode: Replicated**

**Replicas: 1**

**Placement:**

**UpdateConfig:**

**Parallelism: 1**

**On failure: pause**

**Monitoring Period: 5s**

**Max failure ratio: 0**

**Update order: stop-first**

**RollbackConfig:**

**Parallelism: 1**

**On failure: pause**

**Monitoring Period: 5s**

**Max failure ratio: 0**

**Rollback order: stop-first**

**ContainerSpec:**

**Image: tomcat:latest@sha256:026c0a762c65400d5a67c068936540c9203143331b01b4fc85d8af8f8b43272b**

**Resources:**

**Endpoint Mode: vip**

**Ports:**

**PublishedPort = 80**

**Protocol = tcp**

**TargetPort = 80**

**PublishMode = ingress**

