

# Docker Cheat Sheet by DevOps G

→ docker version ( Provides docker version information)

```
[root@ip-172-31-28-147 ~]# docker version
Client:
 Version:      17.05.0-ce
 API version:  1.29
 Go version:   go1.7.5
 Git commit:   89658be
 Built:        Thu May  4 22:06:25 2017
 OS/Arch:      linux/amd64

Server:
 Version:      17.05.0-ce
 API version:  1.29 (minimum version 1.12)
 Go version:   go1.7.5
 Git commit:   89658be
 Built:        Thu May  4 22:06:25 2017
 OS/Arch:      linux/amd64
 Experimental: false
[root@ip-172-31-28-147 ~]# docker --version
Docker version 17.05.0-ce, build 89658be
[root@ip-172-31-28-147 ~]# docker -v
Docker version 17.05.0-ce, build 89658be
[root@ip-172-31-28-147 ~]#
```

→ docker ps (Lists the running container)

```
[root@ip-172-31-28-147 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
6619028b712a       nginx              "nginx -g 'daemon ..." 3 seconds ago       Up 2 seconds       80/tcp, 0.0.0.0:8018->8080/tcp dazling_newton

[root@ip-172-31-28-147 ~]# docker stop 6619028b712a
6619028b712a

[root@ip-172-31-28-147 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
6619028b712a       nginx              "nginx -g 'daemon ..." 26 seconds ago      Exited (0) 6 seconds ago
ewton

[root@ip-172-31-28-147 ~]#
```

→ docker ps -a (this lists all containers either stopped or in running condition)

→ docker ps -l (this is used to show the latest docker container created or modified)

```
[root@ip-172-31-28-147 ~]# docker ps -l
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
6619028b712a       nginx              "nginx -g 'daemon ..." 2 minutes ago      Exited (0) 2 minutes ago
ewton

[root@ip-172-31-28-147 ~]#
```

→ docker images (Lists all available image in host OS in which docker engine is installed)

```
[root@ip-172-31-28-147 ~]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
nginx                latest             5a3221f0137b       45 hours ago       126MB

[root@ip-172-31-28-147 ~]#
```

→ docker pull image\_name (Pulls an image from Docker Hub)

```
[root@ip-172-31-28-147 ~]# docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
1ab2bdf9778: Pull complete
a17e64cfe253: Pull complete
e1288088c7a8: Pull complete
Digest: sha256:53ddb41e46de3d63376579acf46f9a41a8d7de33645db47a486de9769201fec9
Status: Downloaded newer image for nginx:latest
```

→ docker run -it -d image\_name (Runs a command in an isolated container [ i=interactive, t=tty means for terminal, d= for daemon or background process or in detached mode])

```
[root@ip-172-31-28-147 ~]# docker run -itd -p 8018:8080 nginx
6619028b712abd5bec25d256303d4a7df8b9835d7091ea509bc6d3f6a7cf8a21
[root@ip-172-31-28-147 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
6619028b712a       nginx              "nginx -g 'daemon ..." 3 seconds ago       Up 2 seconds       80/tcp, 0.0.0.0:8018->8080/tcp dazling_newton
```

→ docker start container\_id (Starts the stopped containers)

```
[root@ip-172-31-28-147 ~]# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
[root@ip-172-31-28-147 ~]# docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
6619028b712a   nginx    "nginx -g 'daemon ...'" 5 minutes ago    Exited (0) 4 minutes ago          dazzling_newton
[root@ip-172-31-28-147 ~]# docker start 6619028b712a
6619028b712a
[root@ip-172-31-28-147 ~]# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
6619028b712a   nginx    "nginx -g 'daemon ...'" 5 minutes ago    Up 2 seconds    80/tcp, 0.0.0.0:8018->8080/tcp    dazzling_newton
[root@ip-172-31-28-147 ~]#
```

→ docker stop container\_id (Stops the running containers)

```
[root@ip-172-31-28-147 ~]# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
6619028b712a   nginx    "nginx -g 'daemon ...'" 5 minutes ago    Up 2 seconds    80/tcp, 0.0.0.0:8018->8080/tcp    dazzling_newton
[root@ip-172-31-28-147 ~]# docker stop 6619028b712a
6619028b712a
[root@ip-172-31-28-147 ~]# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
[root@ip-172-31-28-147 ~]# docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
6619028b712a   nginx    "nginx -g 'daemon ...'" 6 minutes ago    Exited (0) 6 seconds ago          dazzling_newton
[root@ip-172-31-28-147 ~]#
```

→ docker kill container\_id (this will kill the container)

```
[root@ip-172-31-28-147 ~]# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
6619028b712a   nginx    "nginx -g 'daemon ...'" 8 minutes ago    Up 3 seconds    80/tcp, 0.0.0.0:8018->8080/tcp    dazzling_newton
[root@ip-172-31-28-147 ~]# docker kill 6619028b712a
6619028b712a
[root@ip-172-31-28-147 ~]# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
[root@ip-172-31-28-147 ~]# docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
6619028b712a   nginx    "nginx -g 'daemon ...'" 8 minutes ago    Exited (137) 5 seconds ago          dazzling_newton
[root@ip-172-31-28-147 ~]#
```

→ docker attach (Attaches your local input/output/error stream to a running container)

→ docker commit (Creates a new image from the current changed state of the container)

→ docker exec (to go inside the container)

```
[root@ip-172-31-28-147 ~]# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
6619028b712a   nginx    "nginx -g 'daemon ...'" 10 minutes ago    Up About a minute    80/tcp, 0.0.0.0:8018->8080/tcp    dazzling_newton
[root@ip-172-31-28-147 ~]# docker exec -it 6619028b712a bash
root@6619028b712a:/# ls
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
root@6619028b712a:/# #i am inside nginx container
```

→ docker history image\_name (Displays the history of an image)

```
[root@ip-172-31-28-147 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
nginx	latest	5a3221f0137b	46 hours ago	126MB

```
[root@ip-172-31-28-147 ~]# docker history nginx
```

IMAGE	CREATED	CREATED BY	SIZE
5a3221f0137b	46 hours ago	/bin/sh -c #(nop) CMD ["nginx" "-g" "daem...	0B
<missing>	46 hours ago	/bin/sh -c #(nop) STOPSIGNAL SIGTERM	0B
<missing>	46 hours ago	/bin/sh -c #(nop) EXPOSE 80	0B
<missing>	46 hours ago	/bin/sh -c ln -sf /dev/stdout /var/log/ngi...	0B
<missing>	46 hours ago	/bin/sh -c set -x && addgroup --system...	56.8MB
<missing>	46 hours ago	/bin/sh -c #(nop) ENV PKG_RELEASE=1~buster	0B
<missing>	46 hours ago	/bin/sh -c #(nop) ENV NJS_VERSION=0.3.5	0B
<missing>	46 hours ago	/bin/sh -c #(nop) ENV NGINX_VERSION=1.17.3	0B
<missing>	3 days ago	/bin/sh -c #(nop) LABEL maintainer=NGINX ...	0B
<missing>	3 days ago	/bin/sh -c #(nop) CMD ["bash"]	0B
<missing>	3 days ago	/bin/sh -c #(nop) ADD file:330bfb91168adb4...	69.2MB

→ docker info (Shows system-wide information)

```
[root@ip-172-31-28-147 ~]# docker info
```

```
Containers: 1
  Running: 1
  Paused: 0
  Stopped: 0
Images: 1
Server Version: 17.05.0-ce
Storage Driver: overlay
  Backing Filesystem: xfs
  Supports d_type: true
Logging Driver: json-file
Cgroup Driver: cgroupfs
```

→ docker inspect (Finds system-level information about docker containers and images)

```
[root@ip-172-31-28-147 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
6619028b712a	nginx	"nginx -g 'daemon ..."	16 minutes ago	Up 7 minutes	80/tcp, 0.0.0.0:8018->8080/tcp

```
[root@ip-172-31-28-147 ~]# docker inspect 6619028b712a
```

```
[
  {
    "Id": "6619028b712abd5bec25d256303d4a7df8b9835d7091ea509bc6d3f6a7cf8a21",
    "Created": "2019-08-17T18:44:56.917892708Z",
    "Path": "nginx",
    "Args": [
      "-g",
      "daemon off;"
    ],
  },
]
```

→ docker login (Logins to local registry or Docker Hub)

```
[root@ip-172-31-28-147 ~]# docker login
```

Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head e one.

Username: XXXXXXXXXX

Password:

Login **Succeeded**

```
[root@ip-172-31-28-147 ~]#
```

→ docker restart container\_id (Stops and starts a container)

```
[root@ip-172-31-28-147 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
6619028b712a	nginx	"nginx -g 'daemon ..."	20 minutes ago	Up 11 minutes	80/tcp, 0.0.0.0:8018->8080/tcp

```
ling_newton
[root@ip-172-31-28-147 ~]# docker restart 6619028b712a
6619028b712a
[root@ip-172-31-28-147 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
6619028b712a	nginx	"nginx -g 'daemon ..."	20 minutes ago	Up 2 seconds	80/tcp, 0.0.0.0:8018->8080/tcp

```
ling_newton
[root@ip-172-31-28-147 ~]#
```

→ docker rm container\_id ( Remove stopped containers)

```
[root@ip-172-31-28-147 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
6619028b712a	nginx	"nginx -g 'daemon ..."	21 minutes ago	Up About a minute	80/tcp, 0.0.0.0:8018->8080/tcp

```
ling_newton
[root@ip-172-31-28-147 ~]# docker rm -f 6619028b712a
6619028b712a
[root@ip-172-31-28-147 ~]# docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

```
[root@ip-172-31-28-147 ~]#
```

→ docker rmi image\_name (Remove images)

→ docker rmi \$(docker images -q) (this deletes all docker images for which no container is running)

```
[root@ip-172-31-28-147 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
nginx	latest	5a3221f0137b	46 hours ago	126MB
ubuntu	latest	a2a15febcd3	2 days ago	64.2MB

```
[root@ip-172-31-28-147 ~]# docker images -q
5a3221f0137b
a2a15febcd3
[root@ip-172-31-28-147 ~]# docker rmi $(docker images -q)
Untagged: nginx:latest
Untagged: nginx@sha256:53ddb41e46de3d63376579acf46f9a41a8d7de33645db47a486de9769201fec9
Deleted: sha256:5a3221f0137beb960c34b9cf4455424b6210160fd618c5e79401a07d6e5a2ced
Deleted: sha256:9517458cc6efeca05fee894eb0dda7cda7a704a6469e4624792289c9ff4350c0
Deleted: sha256:963e4580fca3bf364728a7df40a2348939c25213d32508c53849108d05764093
Deleted: sha256:1c95c77433e8d7bf0f519c9d8c9ca967e2603f0defbf379130d9a841cca2e28e
Untagged: ubuntu:latest
Untagged: ubuntu@sha256:d1d454df0f579c6be4d8161d227462d69e163a8ff9d20a847533989cf0c94d90
Deleted: sha256:a2a15febcd362f6115e801d37b5e60d6faeedcb9896155e5fe9d754025be12
Deleted: sha256:fdc47e80ad3dbe1767a5c2141442c0c7aa93a14a817357a0d4353cd8ae48ee58
Deleted: sha256:2b4ed599a73ad82b15d4e3488d95af4f387037b90401d63ad6cb433374b3e3d3
Deleted: sha256:8f06e3a624319de717230f0d766dd8922d048441adb9e0387860a8047d000409
Deleted: sha256:6cebfb3abed5fac58d2e792ce8461454e92c245d5312c42118f02e231a73b317f
[root@ip-172-31-28-147 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
------------	-----	----------	---------	------

```
[root@ip-172-31-28-147 ~]#
```

→ docker rmi -f \$(docker images -q) (this deletes all docker images forcefully)

→ docker rm -f container\_id (this will remove running or stopped container forcefully)

```
[root@ip-172-31-28-147 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
d182b1334595	ubuntu	"/bin/bash"	2 seconds ago	Up 2 seconds	

```
[root@ip-172-31-28-147 ~]# docker rm -f d182b1334595
d182b1334595
[root@ip-172-31-28-147 ~]# docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
--------------	-------	---------	---------	--------	-------

```
[root@ip-172-31-28-147 ~]#
```



→ `docker rm -f $(docker ps -a -q)`

```
[root@ip-172-31-28-147 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
ceb7038ec2ed        ubuntu             "/bin/bash"        6 seconds ago       Up 6 seconds
7b64217e237f        nginx             "nginx -g 'daemon ..." 19 seconds ago      Up 19 seconds      80/tcp
[root@ip-172-31-28-147 ~]# docker rm -f $(docker ps -a -q)
ceb7038ec2ed
7b64217e237f
[root@ip-172-31-28-147 ~]# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
[root@ip-172-31-28-147 ~]#
```

→ `docker search image_name` (Searches the Docker Hub for images)

```
[root@ip-172-31-28-147 ~]# docker search jenkins
NAME                DESCRIPTION                STARS     OFFICIAL    AUTOMATED
jenkins             Official Jenkins Docker image 4394      [OK]
jenkins/jenkins     The leading open source automation server 1632
jenkinsci/blueocean https://jenkins.io/projects/blueocean 418
jenkinsci/jenkins   Jenkins Continuous Integration and Deliver... 367
```

→ `docker save image_name -o new_name.tar` ( the new image name extension should be tar and this saves any image to the tar format only the condition is image should be available on host os)

```
[root@ip-172-31-28-147 ~]# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
nginx         latest   5a3221f0137b   2 days ago    126MB
ubuntu        latest   a2a15febcbdf3  2 days ago    64.2MB
[root@ip-172-31-28-147 ~]# docker save nginx -o my_nginx.tar
[root@ip-172-31-28-147 ~]# ls
my_nginx.tar
```

→ `docker load -i new_name.tar` (this will again create the image which we saved in last command, this mainly used for sharing a docker image from one host to another host)

```
[root@ip-172-31-28-147 ~]# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
ubuntu        latest   a2a15febcbdf3  2 days ago    64.2MB
[root@ip-172-31-28-147 ~]# ls
my_nginx.tar
[root@ip-172-31-28-147 ~]# docker load -i my_nginx.tar
1c95c77433e8: Loading layer [=====>] 72.47MB/72.47MB
002a63507c1c: Loading layer [=====>] 57.31MB/57.31MB
12fdf55172df: Loading layer [=====>] 3.584kB/3.584kB
Loaded image: nginx:latest
[root@ip-172-31-28-147 ~]# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
nginx         latest   5a3221f0137b   2 days ago    126MB
ubuntu        latest   a2a15febcbdf3  2 days ago    64.2MB
[root@ip-172-31-28-147 ~]#
```

Similar to image we can use export and import command for saving container as .tar file and converting that .tar file to docker image as shown below: -

→ `docker export my_container -o my_container.tar` (this will save our container in .tar file)

```
[root@ip-172-31-28-147 ~]# docker run -itd nginx
909bc4383c4bbc5e8a7e98e443aacd4b7a1267943445bf08c19c94802f022f3c
[root@ip-172-31-28-147 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
909bc4383c4b	nginx	"nginx -g 'daemon ...'"	5 seconds ago	Up 4 seconds	80/tcp

```
[root@ip-172-31-28-147 ~]# ls
[root@ip-172-31-28-147 ~]# docker export 909bc4383c4b -o my_container.tar
[root@ip-172-31-28-147 ~]# ls
my_container.tar
[root@ip-172-31-28-147 ~]#
```

→ docker import my\_container.tar (this will create the image which was saved as tar in last command but this image will be treated as filesystem as this doesn't contain all the layers or history which we had seen in earlier command of docker load)

```
[root@ip-172-31-28-147 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
nginx	latest	5a3221f0137b	2 days ago	126MB
ubuntu	latest	a2a15febcd3f	2 days ago	64.2MB

```
[root@ip-172-31-28-147 ~]# ls
my_container.tar  my_nginx
[root@ip-172-31-28-147 ~]# docker import my_container.tar
sha256:300c890aa8e9aac0c9651578e009d078abfe4ede41e357ffd885cab0d8b67ccd
[root@ip-172-31-28-147 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
<none>	<none>	300c890aa8e9	3 seconds ago	124MB
nginx	latest	5a3221f0137b	2 days ago	126MB
ubuntu	latest	a2a15febcd3f	2 days ago	64.2MB

```
[root@ip-172-31-28-147 ~]# docker history 300c890aa8e9
```

IMAGE	CREATED	CREATED BY	SIZE	COMMENT
300c890aa8e9	22 seconds ago		124MB	Imported from -

```
[root@ip-172-31-28-147 ~]#
```

What is difference between docker save and docker export?

Ans→ The main difference is docker save/load with image does preserve the image history. Whereas docker export/import with container flattens the image by removing all the history of the container.

```
[root@ip-172-31-28-147 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
nginx	latest	5a3221f0137b	2 days ago	126MB
ubuntu	latest	a2a15febcd3	2 days ago	64.2MB

```
[root@ip-172-31-28-147 ~]# docker save nginx -o my_nginx
```

```
[root@ip-172-31-28-147 ~]# ls
```

```
my_container.tar  my_nginx
```

```
[root@ip-172-31-28-147 ~]# docker rmi -f nginx
```

```
Untagged: nginx:latest
```

```
Deleted: sha256:5a3221f0137beb960c34b9cf4455424b6210160fd618c5e79401a07d6e5a2ced
```

```
Deleted: sha256:9517458cc6efeca05fee894eb0dda7cda7a704a6469e4624792289c9ff4350c0
```

```
Deleted: sha256:963e4580fca3bf364728a7df40a2348939c25213d32508c53849108d05764093
```

```
Deleted: sha256:1c95c77433e8d7bf0f519c9d8c9ca967e2603f0defbf379130d9a841cca2e28e
```

```
[root@ip-172-31-28-147 ~]# docker load -i my_nginx
```

```
1c95c77433e8: Loading layer [=====>] 72.47MB/72.47MB
```

```
002a63507c1c: Loading layer [=====>] 57.31MB/57.31MB
```

```
12fdf55172df: Loading layer [=====>] 3.584kB/3.584kB
```

```
Loaded image: nginx:latest
```

```
[root@ip-172-31-28-147 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
nginx	latest	5a3221f0137b	2 days ago	126MB
ubuntu	latest	a2a15febcd3	2 days ago	64.2MB

```
[root@ip-172-31-28-147 ~]# docker history nginx
```

IMAGE	CREATED	CREATED BY		SIZE
5a3221f0137b	2 days ago	/bin/sh -c #(nop)	CMD ["nginx" "-g" "daem...	0B
<missing>	2 days ago	/bin/sh -c #(nop)	STOPSIGNAL SIGTERM	0B
<missing>	2 days ago	/bin/sh -c #(nop)	EXPOSE 80	0B
<missing>	2 days ago	/bin/sh -c ln -sf	/dev/stdout /var/log/ngi...	0B
<missing>	2 days ago	/bin/sh -c set -x	&& addgroup --system...	56.8MB
<missing>	2 days ago	/bin/sh -c #(nop)	ENV PKG_RELEASE=1~buster	0B
<missing>	2 days ago	/bin/sh -c #(nop)	ENV NJS_VERSION=0.3.5	0B
<missing>	2 days ago	/bin/sh -c #(nop)	ENV NGINX_VERSION=1.17.3	0B
<missing>	3 days ago	/bin/sh -c #(nop)	LABEL maintainer=NGINX ...	0B
<missing>	4 days ago	/bin/sh -c #(nop)	CMD ["bash"]	0B
<missing>	4 days ago	/bin/sh -c #(nop)	ADD file:330bfb91168adb4...	69.2MB

- docker logs container\_id (this will give the logs of the container)
- docker stats container\_id (this will give the resource usage (CPU, RAM, network i/o) of mentioned container)
- docker stats (this will give resource usage of all available container in host OS)
- docker stats container\_id --no-stream (this will show the resource without streaming)
- docker top container\_id (this will display top process running in a container)

```
[root@ip-172-31-28-147 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
0638c48a7690	ubuntu	"/bin/bash"	2 minutes ago	Up 2 minutes	
bb7aebc231c8	nginx	"nginx -g 'daemon ..."	6 minutes ago	Up 6 minutes	80/tcp

```
[root@ip-172-31-28-147 ~]# docker top bb7aebc231c8
```

UID	PID	PPID	C	STIME	TTY
root	2837	2823	0	04:05	pts/1
inx: master process	nginx -g daemon off;				
101	2854	2837	0	04:05	pts/1
inx: worker process					

- docker build -t myimage:version dockerfile\_path (this will build the docker image from docker file)
- docker tag existing\_image:tag\_name new\_image\_name:new\_tag\_name ( this will create an image with the desired image name and tag name)



```
[root@ip-172-31-28-147 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
nginx	latest	5a3221f0137b	2 days ago	126MB

```
[root@ip-172-31-28-147 ~]# docker tag nginx my_nginx:v1
```

```
[root@ip-172-31-28-147 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
my_nginx	v1	5a3221f0137b	2 days ago	126MB
nginx	latest	5a3221f0137b	2 days ago	126MB

```
[root@ip-172-31-28-147 ~]#
```

Note:- for port mapping -p is for user defined port no. and -P is for random port no. which will be assigned by OS

```
[root@ip-172-31-28-147 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
nginx	latest	5a3221f0137b	2 days ago	126MB

```
[root@ip-172-31-28-147 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
52e3d40195eefc1e5a22a4add12b1e222ab92a9b3775590fd019b575faff3307	nginx	"nginx -g 'daemon ...'"	2 seconds ago	Up 2 seconds	80/tcp, 0.0.0.0:8181->8030/tcp	y_benz

```
[root@ip-172-31-28-147 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
52e3d40195ee	nginx	"nginx -g 'daemon ...'"	2 seconds ago	Up 2 seconds	80/tcp, 0.0.0.0:8181->8030/tcp	y_benz

```
[root@ip-172-31-28-147 ~]#
```

But when we uses -P docker randomly assigns any free port to the port used by nginx i.e 80:-

```
[root@ip-172-31-28-147 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
4b27f5a7501a1923e9d3b6cf4347b397826e2e221f6a5ecfb97b899b326e	nginx	"nginx -g 'daemon ...'"	3 seconds ago	Up 2 seconds	0.0.0.0:32768->80/tcp	

```
[root@ip-172-31-28-147 ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
nginx	latest	5a3221f0137b	2 days ago	126MB

```
[root@ip-172-31-28-147 ~]# docker run -itd -P nginx
```

```
[root@ip-172-31-28-147 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
4b27f5a7501a	nginx	"nginx -g 'daemon ...'"	3 seconds ago	Up 2 seconds	0.0.0.0:32768->80/tcp	

```
[root@ip-172-31-28-147 ~]#
```

Docker volume:-

In below snap we can see by default no docker volume is existing:-

```
[root@ip-172-31-28-147 ~]# docker volume ls
DRIVER          VOLUME NAME
[root@ip-172-31-28-147 ~]# cd /var/lib/docker/volumes/
[root@ip-172-31-28-147 volumes]# ls
metadata.db
[root@ip-172-31-28-147 volumes]#
```

Now will create a docker volume name devopsg :-

```
[root@ip-172-31-28-147 volumes]# docker volume create devopsg
devopsg
[root@ip-172-31-28-147 volumes]# pwd
/var/lib/docker/volumes
[root@ip-172-31-28-147 volumes]# ls
devopsg  metadata.db
[root@ip-172-31-28-147 volumes]# docker volume ls
DRIVER          VOLUME NAME
local           devopsg
[root@ip-172-31-28-147 volumes]#
```

```
[root@ip-172-31-28-147 volumes]# ls
devopsg  metadata.db
[root@ip-172-31-28-147 volumes]# cd devopsg/
[root@ip-172-31-28-147 devopsg]# ls
_data
[root@ip-172-31-28-147 devopsg]# docker volume inspect devopsg
[
  {
    "Driver": "local",
    "Labels": {},
    "Mountpoint": "/var/lib/docker/volumes/devopsg/_data",
    "Name": "devopsg",
    "Options": {},
    "Scope": "local"
  }
]
[root@ip-172-31-28-147 devopsg]#
```

```
[root@ip-172-31-28-147 devopsg]# docker volume ls
DRIVER          VOLUME NAME
local           devopsg
[root@ip-172-31-28-147 devopsg]# docker volume rm devopsg
devopsg
[root@ip-172-31-28-147 devopsg]# docker volume ls
DRIVER          VOLUME NAME
[root@ip-172-31-28-147 devopsg]# cd /var/lib/docker/volumes/
[root@ip-172-31-28-147 volumes]# ls
metadata.db
[root@ip-172-31-28-147 volumes]#
```

docker volume --help (this command gives all the available options to be applied with docker volume)

**\*\*>** Difference between docker volume rm and docker volume prune?

Ans→ docker volume rm removes only docker volume of mentioned name whereas docker prune deletes all existing docker volume .

For more live training, sessions, discussions on docker and all devops tools join us at our facebook page “DevOps G”

Facebook link:- <https://www.facebook.com/groups/devopsg/>

Youtube link:- <https://www.youtube.com/channel/UCw8GNopnFTLRnEGiv0fOQIg/videos>

Telegram Group Name:- DevOps G

<https://www.facebook.com/groups/devopsg/>