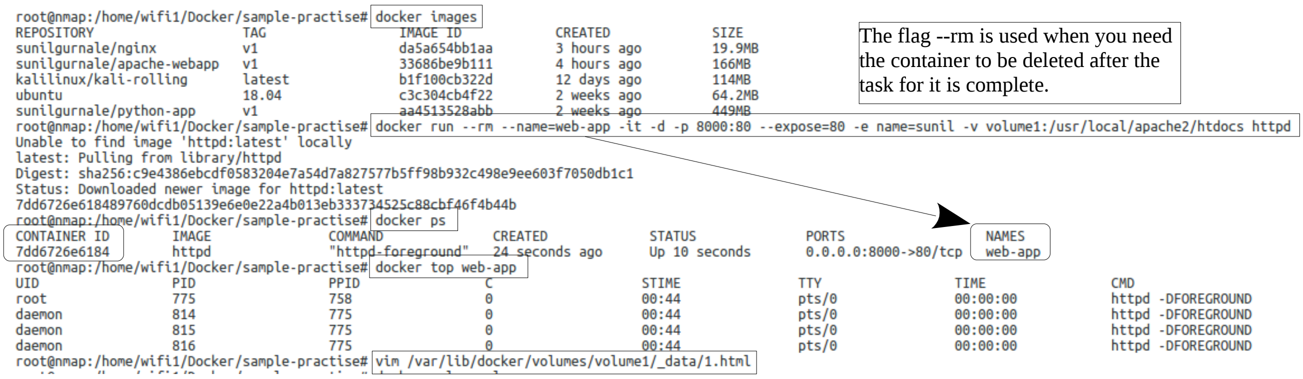
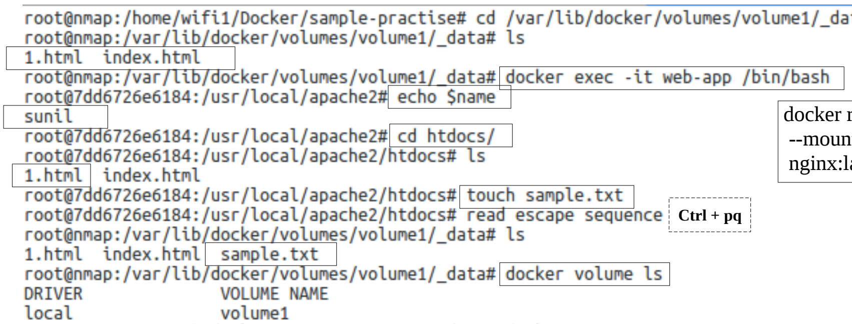


```
root@nmap:/home/wifi1/Docker/sample-practise# docker images
REPOSITORY                  TAG                 IMAGE ID            CREATED             SIZE
sunilgurnale/nginx          v1                  da5a654bb1aa        3 hours ago         19.9MB
sunilgurnale/apache-webapp  v1                  33686be9b111        4 hours ago         166MB
kalilinux/kali-rolling      latest              b1f100cb322d        12 days ago         114MB
ubuntu                      18.04               c3c304cb4f22        2 weeks ago         64.2MB
sunilgurnale/python-app     v1                  aa4513528abb        2 weeks ago         449MB
root@nmap:/home/wifi1/Docker/sample-practise# docker run --rm --name=web-app -it -d -p 8000:80 --expose=80 -e name=sunil -v volume1:/usr/local/apache2/htdocs httpd
Unable to find image 'httpd:latest' locally
latest: Pulling from library/httpd
Digest: sha256:c9e4386ebcdf0583204e7a54d7a827577b5ff98b932c498e9ee603f7050db1c1
Status: Downloaded newer image for httpd:latest
7dd6726e618489760dcdb05139e6e0e22a4b013eb333734525c88cbf46f4b44b
root@nmap:/home/wifi1/Docker/sample-practise# docker ps
CONTAINER ID        IMAGE               COMMAND              CREATED             STATUS              PORTS                  NAMES
7dd6726e6184        httpd               "httpd-foreground"   24 seconds ago      Up 10 seconds       0.0.0.0:8000->80/tcp   web-app
root@nmap:/home/wifi1/Docker/sample-practise# docker top web-app
UID                 PID                 PPID                C                   STIME               TTY                 TIME                CMD
root                775                 758                 0                   00:44               pts/0               00:00:00            httpd -DFOREGROUND
daemon              814                 775                 0                   00:44               pts/0               00:00:00            httpd -DFOREGROUND
daemon              815                 775                 0                   00:44               pts/0               00:00:00            httpd -DFOREGROUND
daemon              816                 775                 0                   00:44               pts/0               00:00:00            httpd -DFOREGROUND
root@nmap:/home/wifi1/Docker/sample-practise# vim /var/lib/docker/volumes/volume1/_data/1.html
```

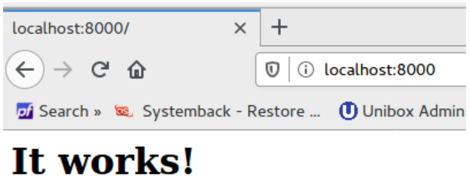
The flag --rm is used when you need the container to be deleted after the task for it is complete.



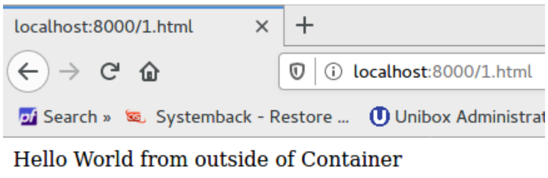
```
root@nmap:/home/wifi1/Docker/sample-practise# cd /var/lib/docker/volumes/volume1/_data/
root@nmap:/var/lib/docker/volumes/volume1/_data# ls
1.html  index.html
root@nmap:/var/lib/docker/volumes/volume1/_data# docker exec -it web-app /bin/bash
root@7dd6726e6184:/usr/local/apache2# echo $name
sunil
root@7dd6726e6184:/usr/local/apache2# cd htdocs/
root@7dd6726e6184:/usr/local/apache2/htdocs# ls
1.html  index.html
root@7dd6726e6184:/usr/local/apache2/htdocs# touch sample.txt
root@7dd6726e6184:/usr/local/apache2/htdocs# read escape sequence
root@nmap:/var/lib/docker/volumes/volume1/_data# ls
1.html  index.html  sample.txt
root@nmap:/var/lib/docker/volumes/volume1/_data# docker volume ls
DRIVER              VOLUME NAME
local               volume1
```

Ctrl + pq

```
docker run -d -it --name devtest \
 --mount type=bind,source="$(pwd)"/target,target=/app,readonly \
 nginx:latest
```

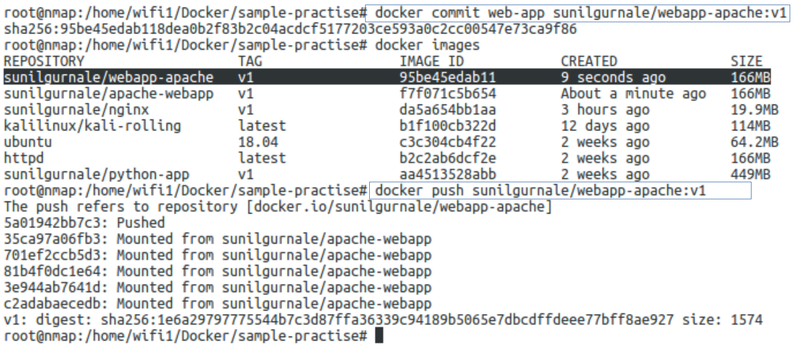


It works!

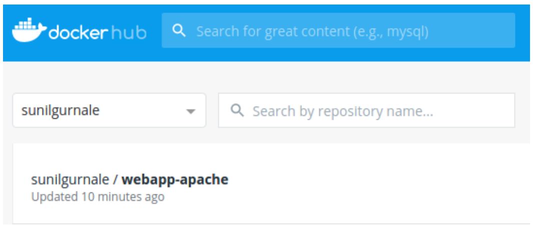


Hello World from outside of Container

Commit and Push:



```
root@nmap:/home/wifi1/Docker/sample-practise# docker commit web-app sunilgurnale/webapp-apache:v1
sha256:95be45edab118dea0b2f83b2c04acdcf5177203ce593a0c2cc00547e73ca9f86
root@nmap:/home/wifi1/Docker/sample-practise# docker images
REPOSITORY                  TAG                 IMAGE ID            CREATED              SIZE
sunilgurnale/webapp-apache  v1                  95be45edab11        9 seconds ago        166MB
sunilgurnale/apache-webapp  v1                  f7f071c5b654        About a minute ago   166MB
sunilgurnale/nginx          v1                  da5a654bb1aa        3 hours ago          19.9MB
kalilinux/kali-rolling      latest              b1f100cb322d        12 days ago          114MB
ubuntu                      18.04               c3c304cb4f22        2 weeks ago          64.2MB
httpd                       latest              b2c2ab6dcf2e        2 weeks ago          166MB
sunilgurnale/python-app     v1                  aa4513528abb        2 weeks ago          449MB
root@nmap:/home/wifi1/Docker/sample-practise# docker push sunilgurnale/webapp-apache:v1
The push refers to repository [docker.io/sunilgurnale/webapp-apache]
5a01942bb7c3: Pushed
35ca97a06fb3: Mounted from sunilgurnale/apache-webapp
701ef2ccb5d3: Mounted from sunilgurnale/apache-webapp
81b4f0dc1e64: Mounted from sunilgurnale/apache-webapp
3e944ab7641d: Mounted from sunilgurnale/apache-webapp
c2adabaecedb: Mounted from sunilgurnale/apache-webapp
v1: digest: sha256:1e6a29797775544b7c3d87ffa36339c94189b5065e7dbcdffdeee77bff8ae927 size: 1574
root@nmap:/home/wifi1/Docker/sample-practise# █
```

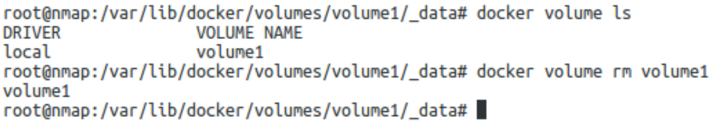


docker hub
Search for great content (e.g., mysql)
sunilgurnale
Search by repository name...

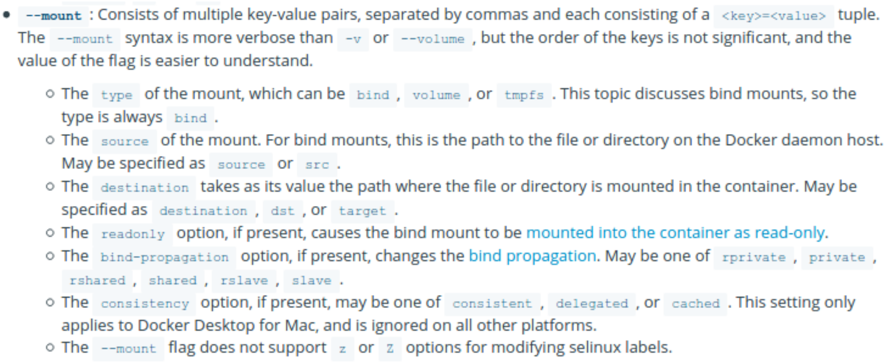
sunilgurnale / **webapp-apache**
Updated 10 minutes ago

Remove / Clean :

- docker kill $(docker ps -aq)
- docker rm $(docker ps -aq)
- docker volume rm <volume_name>
- docker system prune

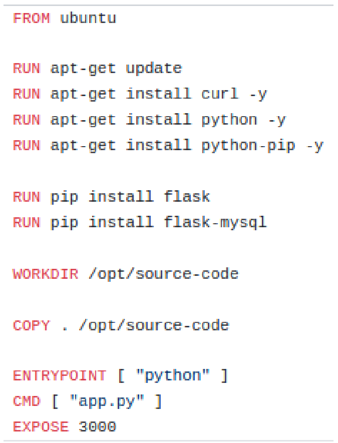


```
root@nmap:/var/lib/docker/volumes/volume1/_data# docker volume ls
DRIVER              VOLUME NAME
local               volume1
root@nmap:/var/lib/docker/volumes/volume1/_data# docker volume rm volume1
volume1
root@nmap:/var/lib/docker/volumes/volume1/_data# █
```



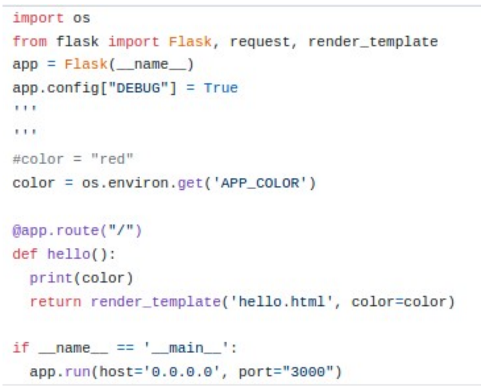
- `--mount` : Consists of multiple key-value pairs, separated by commas and each consisting of a `<key>=<value>` tuple. The `--mount` syntax is more verbose than `-v` or `--volume` , but the order of the keys is not significant, and the value of the flag is easier to understand.
  - The `type` of the mount, which can be `bind` , `volume` , or `tmpfs` . This topic discusses bind mounts, so the type is always `bind` .
  - The `source` of the mount. For bind mounts, this is the path to the file or directory on the Docker daemon host. May be specified as `source` or `src` .
  - The `destination` takes as its value the path where the file or directory is mounted in the container. May be specified as `destination` , `dst` , or `target` .
  - The `readonly` option, if present, causes the bind mount to be mounted into the container as read-only.
  - The `bind-propagation` option, if present, changes the bind propagation. May be one of `rprivate` , `private` , `rshared` , `shared` , `rslave` , `slave` .
  - The `consistency` option, if present, may be one of `consistent` , `delegated` , or `cached` . This setting only applies to Docker Desktop for Mac, and is ignored on all other platforms.
  - The `--mount` flag does not support `z` or `Z` options for modifying selinux labels.

DockerFile

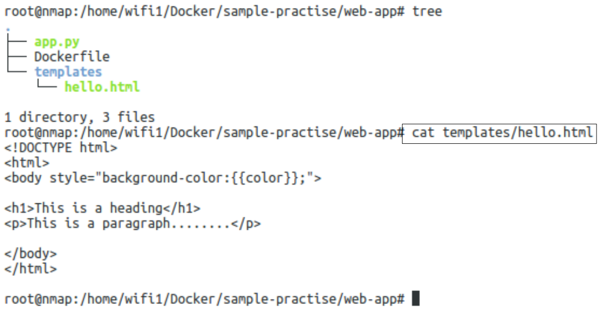


```
FROM ubuntu

RUN apt-get update
RUN apt-get install curl -y
RUN apt-get install python -y
RUN apt-get install python-pip -y

RUN pip install flask
RUN pip install flask-mysql

WORKDIR /opt/source-code

COPY . /opt/source-code

ENTRYPOINT [ "python" ]
CMD [ "app.py" ]
EXPOSE 3000
```

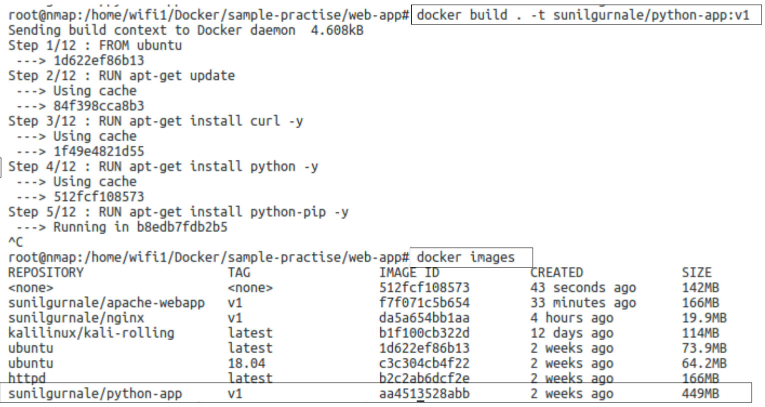
app.py



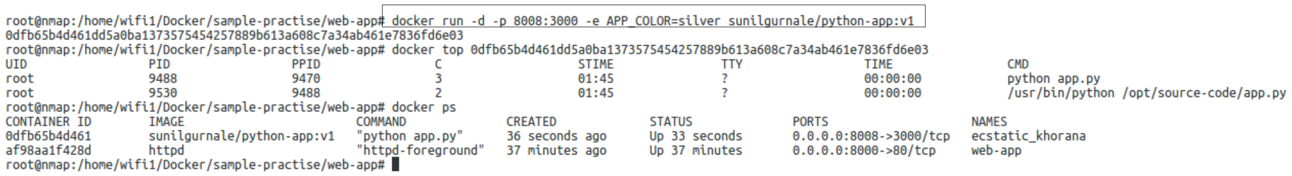
```
import os
from flask import Flask, request, render_template
app = Flask(__name__)
app.config["DEBUG"] = True
'''
'''
#color = "red"
color = os.environ.get('APP_COLOR')

@app.route("/")
def hello():
  print(color)
  return render_template('hello.html', color=color)

if __name__ == '__main__':
  app.run(host='0.0.0.0', port="3000")
```



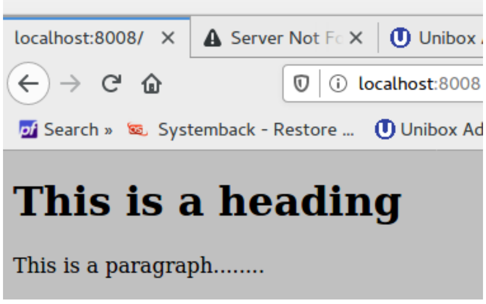
```
root@nmap:/home/wifi1/Docker/sample-practise/web-app# tree
.
├── app.py
├── Dockerfile
└── templates
    └── hello.html

1 directory, 3 files
root@nmap:/home/wifi1/Docker/sample-practise/web-app# cat templates/hello.html
<!DOCTYPE html>
<html>
<body style="background-color:{{color}};">

<h1>This is a heading</h1>
<p>This is a paragraph........</p>

</body>
</html>

root@nmap:/home/wifi1/Docker/sample-practise/web-app# █
```



```
root@nmap:/home/wifi1/Docker/sample-practise/web-app# docker build . -t sunilgurnale/python-app:v1
Sending build context to Docker daemon  4.608kB
Step 1/12 : FROM ubuntu
 ---> 1d622ef86b13
Step 2/12 : RUN apt-get update
 ---> Using cache
 ---> 84f398cca8b3
Step 3/12 : RUN apt-get install curl -y
 ---> Using cache
 ---> 1f49e4821d55
Step 4/12 : RUN apt-get install python -y
 ---> Using cache
 ---> 512fcf108573
Step 5/12 : RUN apt-get install python-pip -y
 ---> Running in b8edb7fdb2b5
^C
root@nmap:/home/wifi1/Docker/sample-practise/web-app# docker images
REPOSITORY                  TAG         IMAGE ID        CREATED           SIZE
<none>                      <none>      512fcf108573    43 seconds ago    142MB
sunilgurnale/apache-webapp  v1          f7f071c5b654    33 minutes ago    166MB
sunilgurnale/nginx          v1          da5a654bb1aa    4 hours ago       19.9MB
kalilinux/kali-rolling      latest      b1f100cb322d    12 days ago       114MB
ubuntu                      latest      1d622ef86b13    2 weeks ago       73.9MB
ubuntu                      18.04       c3c304cb4f22    2 weeks ago       64.2MB
httpd                       latest      b2c2ab6dcf2e    2 weeks ago       166MB
sunilgurnale/python-app     v1          aa4513528abb    2 weeks ago       449MB
```



```
root@nmap:/home/wifi1/Docker/sample-practise/web-app# docker run -d -p 8008:3000 -e APP_COLOR=silver sunilgurnale/python-app:v1
0dfb65b4d461dd5a0ba1373575454257889b613a608c7a34ab461e7836fd6e03
root@nmap:/home/wifi1/Docker/sample-practise/web-app# docker top 0dfb65b4d461dd5a0ba1373575454257889b613a608c7a34ab461e7836fd6e03
UID         PID         PPID        C           STIME       TTY         TIME        CMD
root        9488        9470        3           01:45       ?           00:00:00    python app.py
root        9530        9488        2           01:45       ?           00:00:00    /usr/bin/python /opt/source-code/app.py
root@nmap:/home/wifi1/Docker/sample-practise/web-app# docker ps
CONTAINER ID   IMAGE                        COMMAND               CREATED           STATUS           PORTS                      NAMES
0dfb65b4d461   sunilgurnale/python-app:v1   "python app.py"       36 seconds ago    Up 33 seconds    0.0.0.0:8008->3000/tcp     ecstatic_khorana
af98aa1f428d   httpd                        "httpd-foreground"    37 minutes ago    Up 37 minutes    0.0.0.0:8000->80/tcp       web-app
root@nmap:/home/wifi1/Docker/sample-practise/web-app# █
```

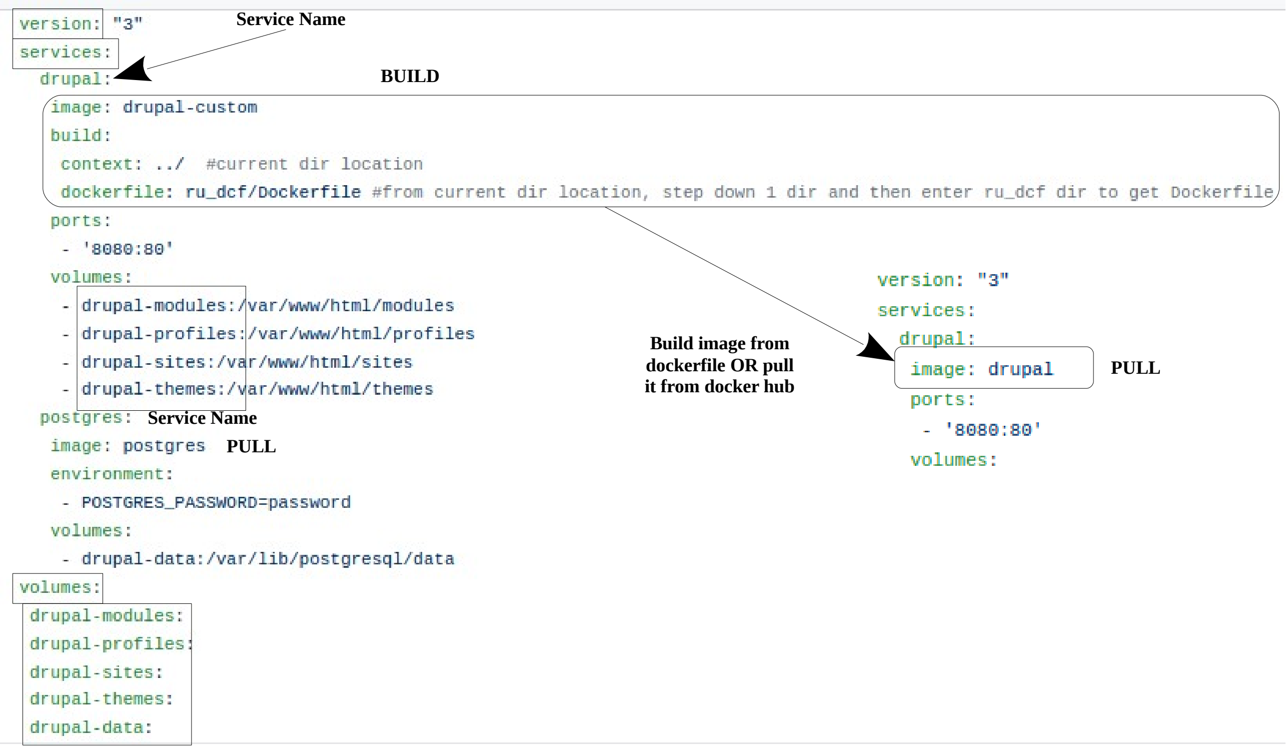


This is a heading

This is a paragraph........

- -p = publish
- -d = detach mode
- -e = env_variable
- run –rm .....
- docker kill $(docker ps -aq)
- docker rm $(docker ps -aq)
- docker system prune

Docker-Compose: Version Present 1,2,3



```
version: "3"
services:
  drupal:
    image: drupal-custom
    build:
     context: ../  #current dir location
     dockerfile: ru_dcf/Dockerfile #from current dir location, step down 1 dir and then enter ru_dcf dir to get Dockerfile
    ports:
     - '8080:80'
    volumes:
     - drupal-modules:/var/www/html/modules
     - drupal-profiles:/var/www/html/profiles
     - drupal-sites:/var/www/html/sites
     - drupal-themes:/var/www/html/themes
  postgres:
    image: postgres
    environment:
     - POSTGRES_PASSWORD=password
    volumes:
     - drupal-data:/var/lib/postgresql/data
volumes:
  drupal-modules:
  drupal-profiles:
  drupal-sites:
  drupal-themes:
  drupal-data:
```

```
version: "3"
services:
  drupal:
    image: drupal
    ports:
     - '8080:80'
    volumes:
```

Service Name

BUILD

Build image from dockerfile OR pull it from docker hub

PULL

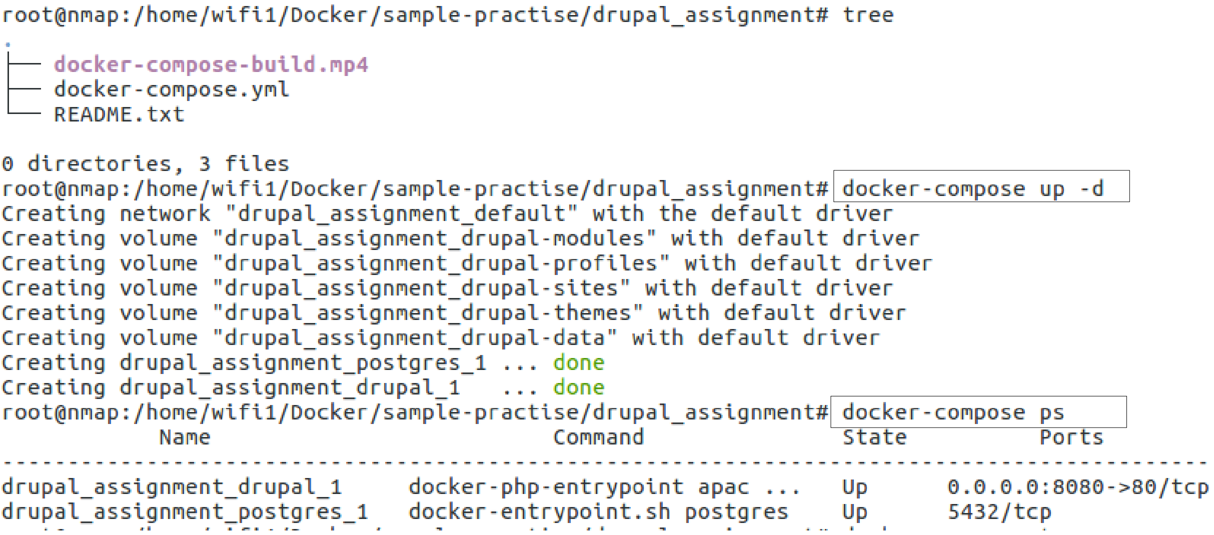
Service Name

PULL

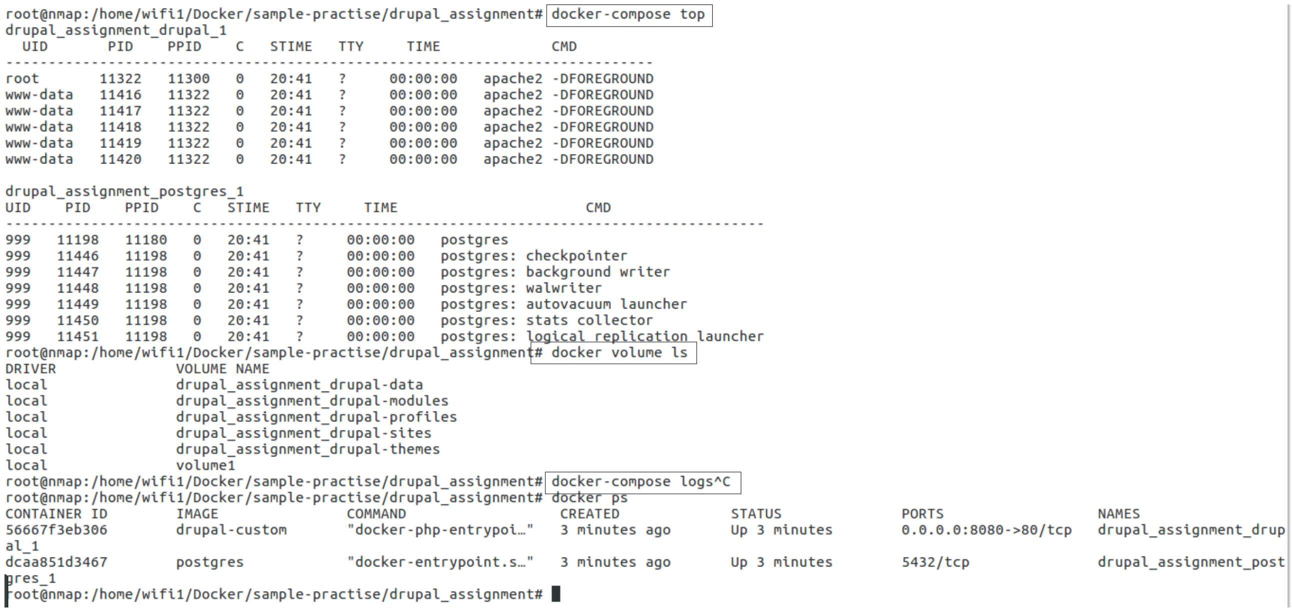
Install Docker-compose :

- sudo curl -L "https://github.com/docker/compose/releases/download/1.25.5/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
- sudo chmod +x /usr/local/bin/docker-compose
- docker-compose –version
- https://docs.docker.com/compose/install/ -- installation steps

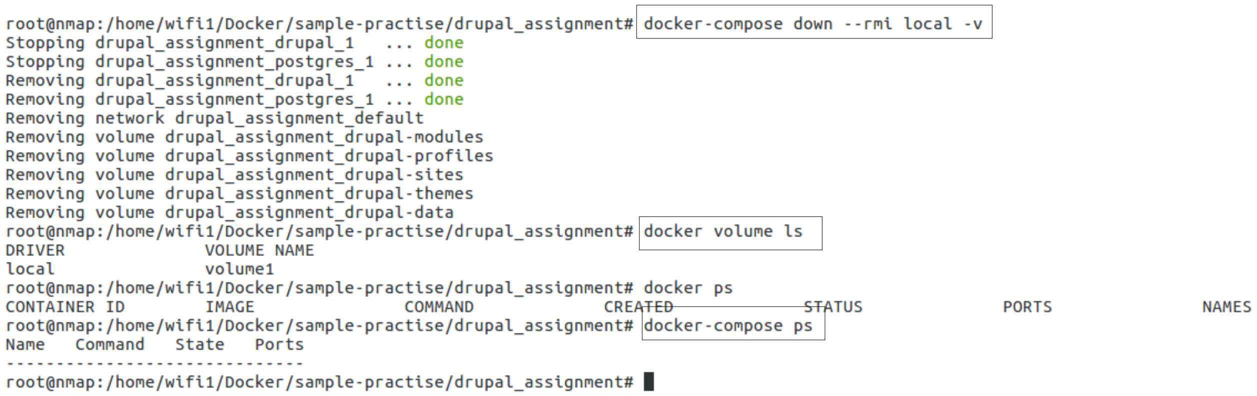
Executing steps:



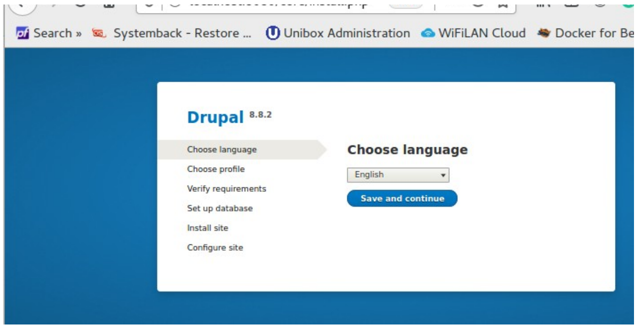
```
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# tree
.
├── docker-compose-build.mp4
├── docker-compose.yml
└── README.txt

0 directories, 3 files
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# docker-compose up -d
Creating network "drupal_assignment_default" with the default driver
Creating volume "drupal_assignment_drupal-modules" with default driver
Creating volume "drupal_assignment_drupal-profiles" with default driver
Creating volume "drupal_assignment_drupal-sites" with default driver
Creating volume "drupal_assignment_drupal-themes" with default driver
Creating volume "drupal_assignment_drupal-data" with default driver
Creating drupal_assignment_postgres_1 ... done
Creating drupal_assignment_drupal_1   ... done
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# docker-compose ps
            Name                          Command               State          Ports
-------------------------------------------------------------------------------------------
drupal_assignment_drupal_1     docker-php-entrypoint apac ...   Up      0.0.0.0:8080->80/tcp
drupal_assignment_postgres_1   docker-entrypoint.sh postgres    Up      5432/tcp
```



```
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# docker-compose top
drupal_assignment_drupal_1
  UID       PID    PPID    C   STIME    TTY     TIME              CMD
--------------------------------------------------------------------------
root       11322   11300   0   20:41    ?     00:00:00   apache2 -DFOREGROUND
www-data   11416   11322   0   20:41    ?     00:00:00   apache2 -DFOREGROUND
www-data   11417   11322   0   20:41    ?     00:00:00   apache2 -DFOREGROUND
www-data   11418   11322   0   20:41    ?     00:00:00   apache2 -DFOREGROUND
www-data   11419   11322   0   20:41    ?     00:00:00   apache2 -DFOREGROUND
www-data   11420   11322   0   20:41    ?     00:00:00   apache2 -DFOREGROUND

drupal_assignment_postgres_1
UID    PID    PPID    C   STIME   TTY     TIME                      CMD
---------------------------------------------------------------------------------------
999    11198   11180   0   20:41   ?     00:00:00   postgres
999    11446   11198   0   20:41   ?     00:00:00   postgres: checkpointer
999    11447   11198   0   20:41   ?     00:00:00   postgres: background writer
999    11448   11198   0   20:41   ?     00:00:00   postgres: walwriter
999    11449   11198   0   20:41   ?     00:00:00   postgres: autovacuum launcher
999    11450   11198   0   20:41   ?     00:00:00   postgres: stats collector
999    11451   11198   0   20:41   ?     00:00:00   postgres: logical replication launcher
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# docker volume ls
DRIVER              VOLUME NAME
local               drupal_assignment_drupal-data
local               drupal_assignment_drupal-modules
local               drupal_assignment_drupal-profiles
local               drupal_assignment_drupal-sites
local               drupal_assignment_drupal-themes
local               volume1
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# docker-compose logs^C
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS                  NAMES
56667f3eb306        drupal-custom       "docker-php-entrypoi…"   3 minutes ago       Up 3 minutes        0.0.0.0:8080->80/tcp   drupal_assignment_drup
al_1
dcaa851d3467        postgres            "docker-entrypoint.s…"   3 minutes ago       Up 3 minutes        5432/tcp               drupal_assignment_post
gres_1
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# 
```



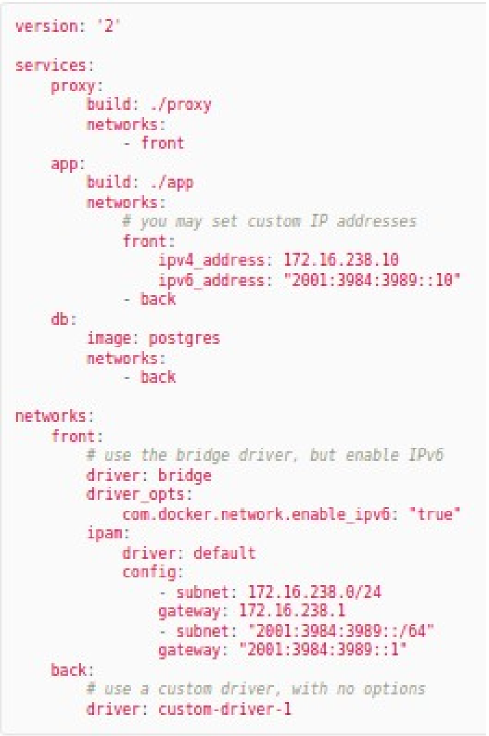
```
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# docker-compose down --rmi local -v
Stopping drupal_assignment_drupal_1   ... done
Stopping drupal_assignment_postgres_1 ... done
Removing drupal_assignment_drupal_1   ... done
Removing drupal_assignment_postgres_1 ... done
Removing network drupal_assignment_default
Removing volume drupal_assignment_drupal-modules
Removing volume drupal_assignment_drupal-profiles
Removing volume drupal_assignment_drupal-sites
Removing volume drupal_assignment_drupal-themes
Removing volume drupal_assignment_drupal-data
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# docker volume ls
DRIVER              VOLUME NAME
local               volume1
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# docker-compose ps
Name   Command   State   Ports
------------------------------
root@nmap:/home/wifi1/Docker/sample-practise/drupal_assignment# 
```



- docker network ls
- docker network inspect bridge
- brctl show

Best Example:

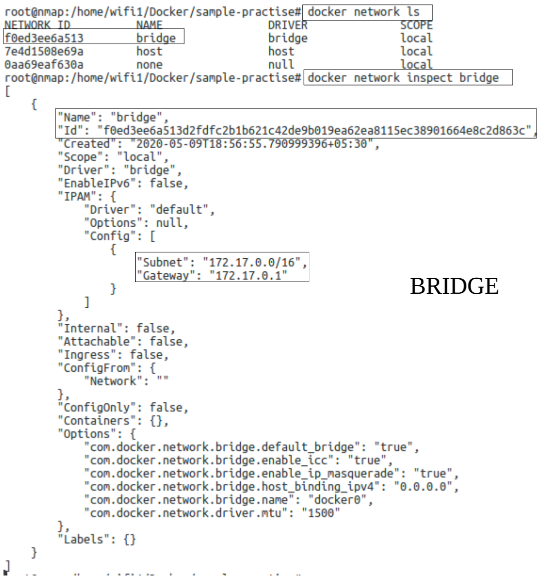
https://docker-k8s-lab.readthedocs.io/en/latest/docker/docker-compose.html



```
version: '2'

services:
    proxy:
        build: ./proxy
        networks:
            - front
    app:
        build: ./app
        networks:
            # you may set custom IP addresses
            front:
                ipv4_address: 172.16.238.10
                ipv6_address: "2001:3984:3989::10"
            - back
    db:
        image: postgres
        networks:
            - back

networks:
    front:
        # use the bridge driver, but enable IPv6
        driver: bridge
        driver_opts:
            com.docker.network.enable_ipv6: "true"
        ipam:
            driver: default
            config:
                - subnet: 172.16.238.0/24
                gateway: 172.16.238.1
                - subnet: "2001:3984:3989::/64"
                gateway: "2001:3984:3989::1"
    back:
        # use a custom driver, with no options
        driver: custom-driver-1
```

Networking:

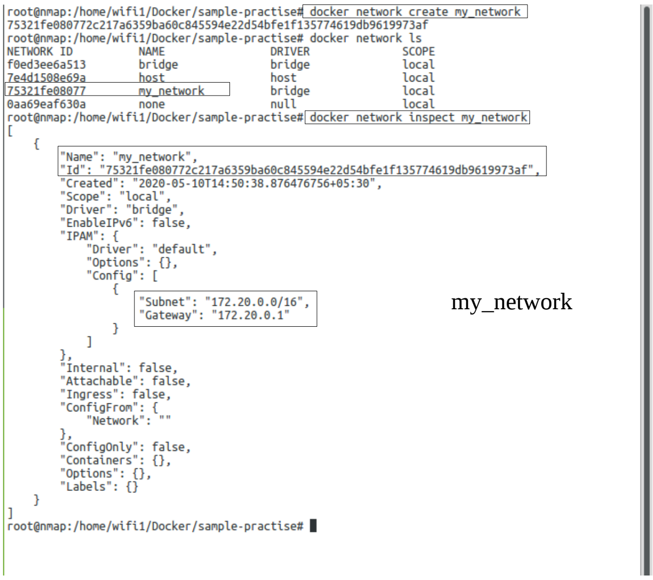
bridge



```
root@nmap:/home/wifi1/Docker/sample-practise# docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
f0ed3ee6a513        bridge              bridge              local
7e4d1508e69a        host                host                local
0aa69eaf630a        none                null                local
root@nmap:/home/wifi1/Docker/sample-practise# docker network inspect bridge
[
    {
        "Name": "bridge",
        "Id": "f0ed3ee6a513d2fdfc2b1b621c42de9b019ea62ea8115ec38901664e8c2d863c",
        "Created": "2020-05-09T18:56:55.790999396+05:30",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "172.17.0.0/16",
                    "Gateway": "172.17.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "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": {}
    }
]
```

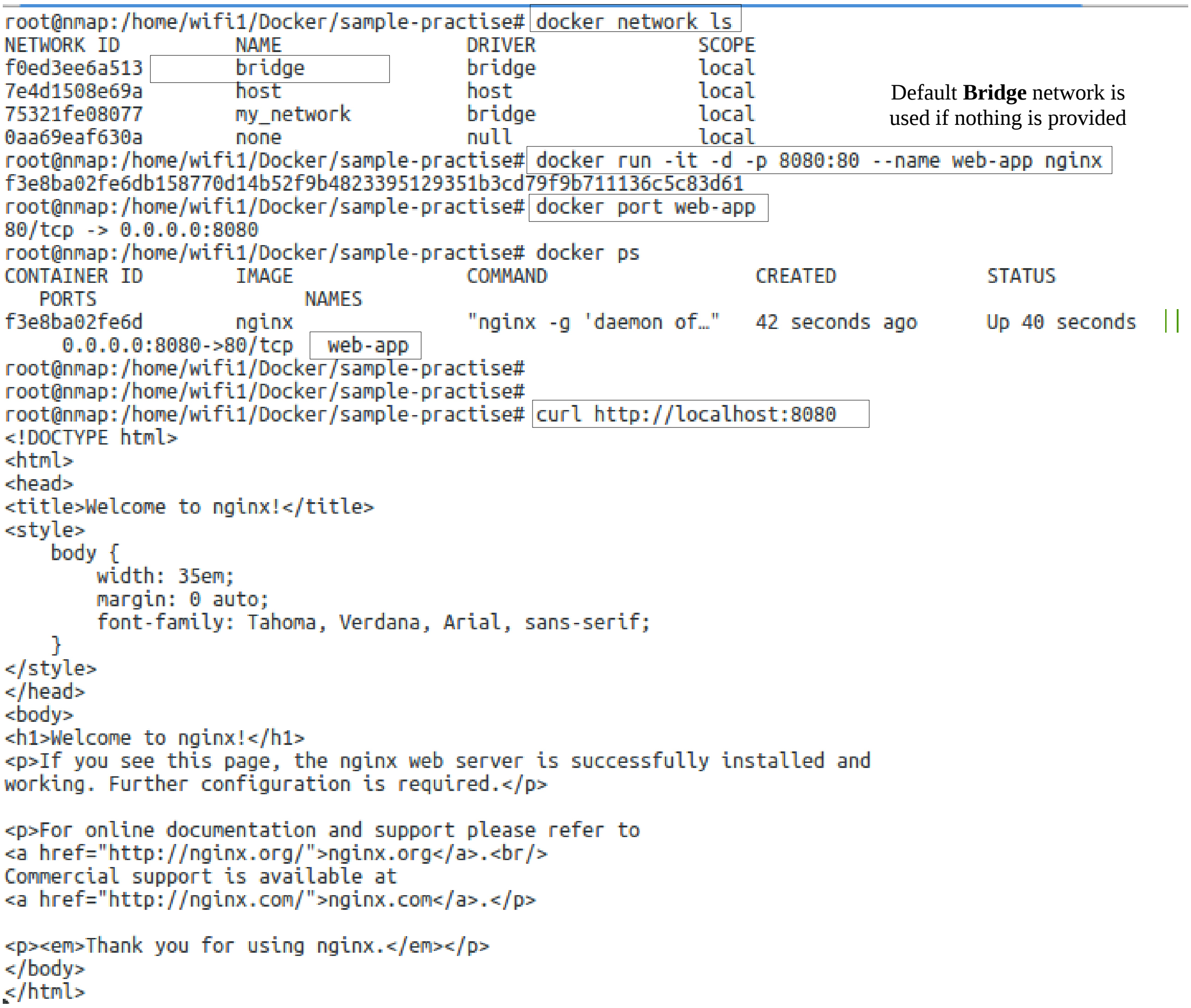
BRIDGE

my_network



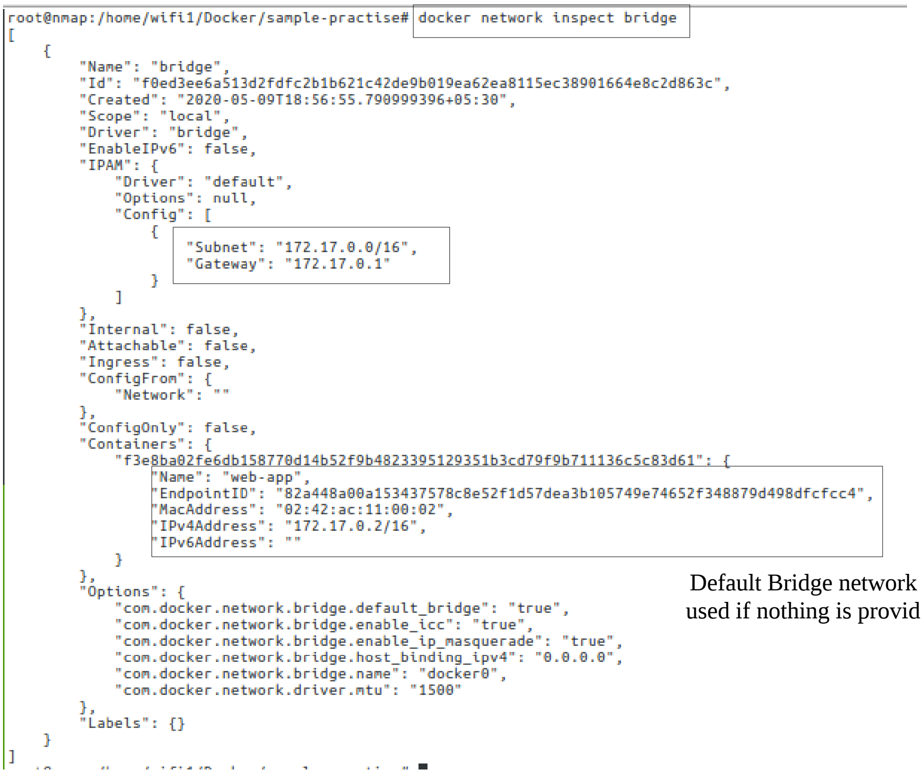
```
root@nmap:/home/wifi1/Docker/sample-practise# docker network create my_network
75321fe080772c217a6359ba60c845594e22d54bfe1f135774619db9619973af
root@nmap:/home/wifi1/Docker/sample-practise# docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
f0ed3ee6a513        bridge              bridge              local
7e4d1508e69a        host                host                local
75321fe08077        my_network          bridge              local
0aa69eaf630a        none                null                local
root@nmap:/home/wifi1/Docker/sample-practise# docker network inspect my_network
[
    {
        "Name": "my_network",
        "Id": "75321fe080772c217a6359ba60c845594e22d54bfe1f135774619db9619973af",
        "Created": "2020-05-10T14:50:38.876476756+05:30",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.20.0.0/16",
                    "Gateway": "172.20.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "Options": {},
        "Labels": {}
    }
]
root@nmap:/home/wifi1/Docker/sample-practise# 
```

my_network



```
root@nmap:/home/wifi1/Docker/sample-practise# docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
f0ed3ee6a513        bridge              bridge              local
7e4d1508e69a        host                host                local
75321fe08077        my_network          bridge              local
0aa69eaf630a        none                null                local
root@nmap:/home/wifi1/Docker/sample-practise# docker run -it -d -p 8080:80 --name web-app nginx
f3e8ba02fe6db158770d14b52f9b4823395129351b3cd79f9b711136c5c83d61
root@nmap:/home/wifi1/Docker/sample-practise# docker port web-app
80/tcp -> 0.0.0.0:8080
root@nmap:/home/wifi1/Docker/sample-practise# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS
   PORTS                  NAMES
f3e8ba02fe6d        nginx               "nginx -g 'daemon of…"   42 seconds ago      Up 40 seconds
     0.0.0.0:8080->80/tcp   web-app
root@nmap:/home/wifi1/Docker/sample-practise#
root@nmap:/home/wifi1/Docker/sample-practise#
root@nmap:/home/wifi1/Docker/sample-practise# curl http://localhost:8080
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

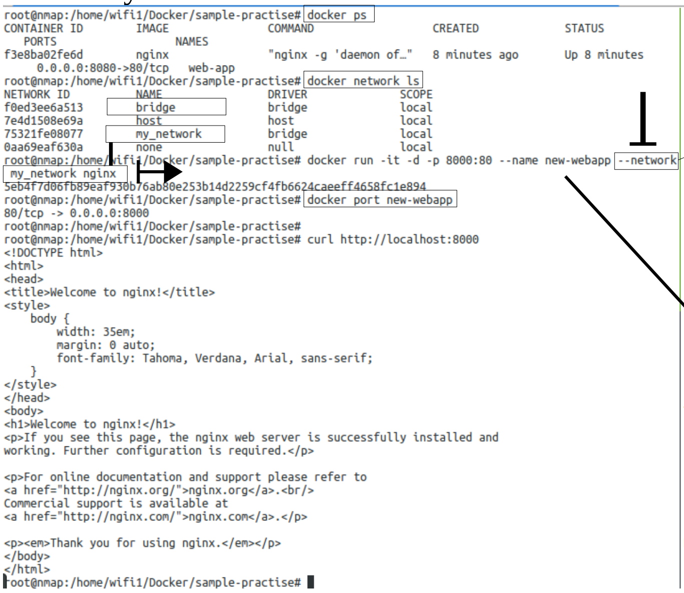
Default **Bridge** network is used if nothing is provided



```
root@nmap:/home/wifi1/Docker/sample-practise# docker network inspect bridge
[
    {
        "Name": "bridge",
        "Id": "f0ed3ee6a513d2fdfc2b1b621c42de9b019ea62ea8115ec38901664e8c2d863c",
        "Created": "2020-05-09T18:56:55.790999396+05:30",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "172.17.0.0/16",
                    "Gateway": "172.17.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "f3e8ba02fe6db158770d14b52f9b4823395129351b3cd79f9b711136c5c83d61": {
                "Name": "web-app",
                "EndpointID": "82a448a00a153437578c8e52f1d57dea3b105749e74652f348879d498dfcfcc4",
                "MacAddress": "02:42:ac:11:00:02",
                "IPv4Address": "172.17.0.2/16",
                "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": {}
    }
]
root@nmap:/home/wifi1/Docker/sample-practise#
```

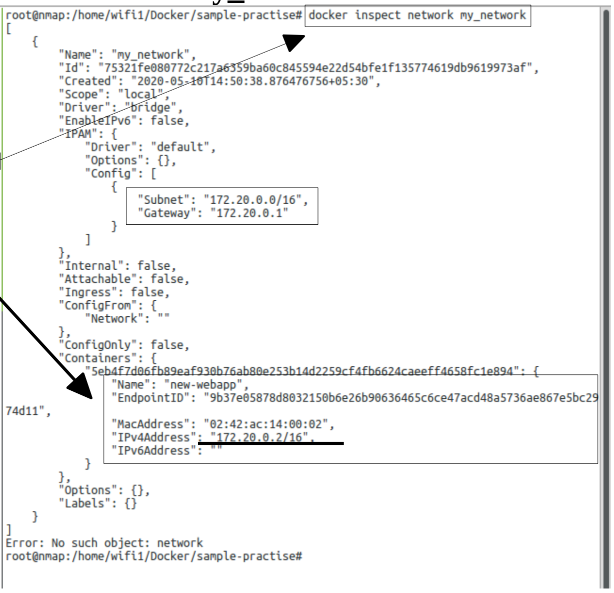
Default Bridge network is used if nothing is provided

Use newly created network:

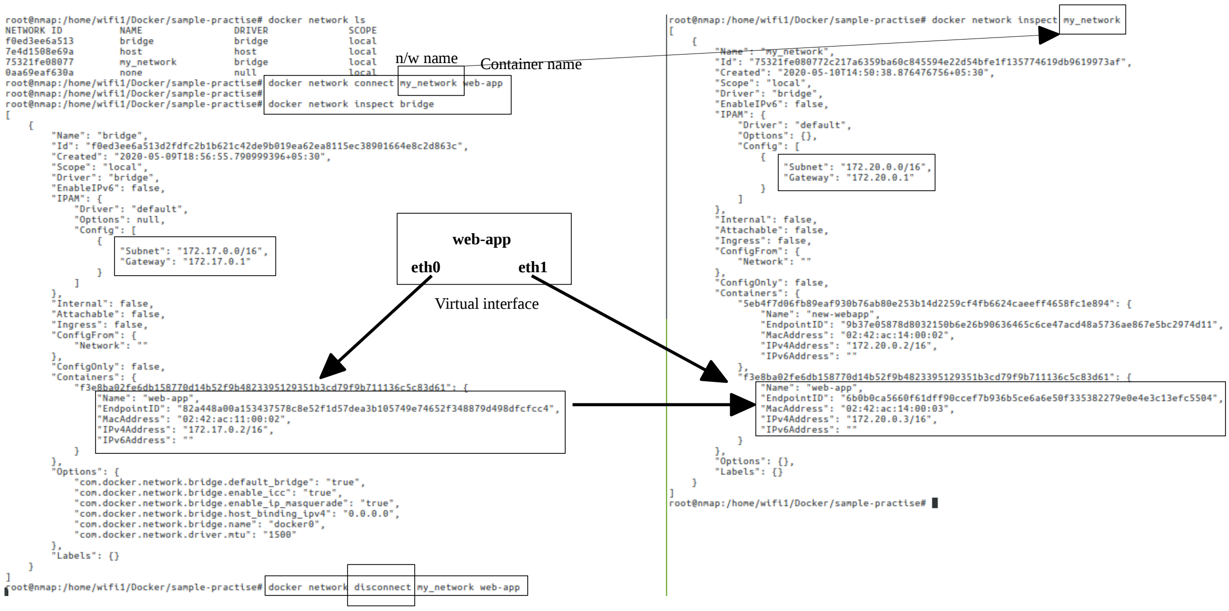


```
root@nmap:/home/wifi1/Docker/sample-practise# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS
   PORTS                  NAMES
f3e8ba02fe6d        nginx               "nginx -g 'daemon of…"   8 minutes ago       Up 8 minutes
     0.0.0.0:8080->80/tcp   web-app
root@nmap:/home/wifi1/Docker/sample-practise# docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
f0ed3ee6a513        bridge              bridge              local
7e4d1508e69a        host                host                local
75321fe08077        my_network          bridge              local
0aa69eaf630a        none                null                local
root@nmap:/home/wifi1/Docker/sample-practise# docker run -it -d -p 8000:80 --name new-webapp --network
 my_network nginx
5eb4f7d06fb89eaf930b76ab80e253b14d2259cf4fb6624caeeff4658fc1e894
root@nmap:/home/wifi1/Docker/sample-practise# docker port new-webapp
80/tcp -> 0.0.0.0:8000
root@nmap:/home/wifi1/Docker/sample-practise#
root@nmap:/home/wifi1/Docker/sample-practise# curl http://localhost:8000
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
root@nmap:/home/wifi1/Docker/sample-practise#
```

docker network create **my_network**

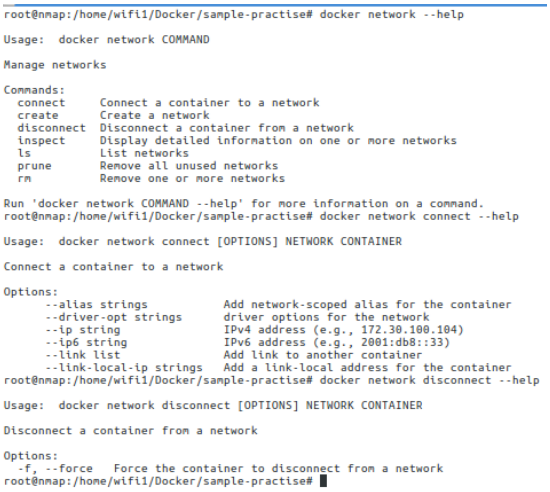


```
root@nmap:/home/wifi1/Docker/sample-practise# docker inspect network my_network
[
    {
        "Name": "my_network",
        "Id": "75321fe080772c217a6359ba60c845594e22d54bfe1f135774619db9619973af",
        "Created": "2020-05-10T14:50:38.876476756+05:30",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.20.0.0/16",
                    "Gateway": "172.20.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "5eb4f7d06fb89eaf930b76ab80e253b14d2259cf4fb6624caeeff4658fc1e894": {
                "Name": "new-webapp",
                "EndpointID": "9b37e05878d8032150b6e26b90636465c6ce47acd48a5736ae867e5bc29
74d11",
                "MacAddress": "02:42:ac:14:00:02",
                "IPv4Address": "172.20.0.2/16",
                "IPv6Address": ""
            }
        },
        "Options": {},
        "Labels": {}
    }
]
Error: No such object: network
root@nmap:/home/wifi1/Docker/sample-practise#
```



```
root@nmap:/home/wifi1/Docker/sample-practise# docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
f0ed3ee6a513        bridge              bridge              local
7e4d1508e69a        host                host                local
75321fe08077        my_network          bridge              local
0aa69eaf630a        none                null                local
root@nmap:/home/wifi1/Docker/sample-practise# docker network connect my_network web-app
root@nmap:/home/wifi1/Docker/sample-practise#
root@nmap:/home/wifi1/Docker/sample-practise# docker network inspect bridge
[
    {
        "Name": "bridge",
        "Id": "f0ed3ee6a513d2fdfc2b1b621c42de9b019ea62ea8115ec38901664e8c2d863c",
        "Created": "2020-05-09T18:56:55.790999396+05:30",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "172.17.0.0/16",
                    "Gateway": "172.17.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "f3e8ba02fe6db158770d14b52f9b4823395129351b3cd79f9b711136c5c83d61": {
                "Name": "web-app",
                "EndpointID": "82a448a00a153437578c8e52f1d57dea3b105749e74652f348879d498dfcfcc4",
                "MacAddress": "02:42:ac:11:00:02",
                "IPv4Address": "172.17.0.2/16",
                "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": {}
    }
]
root@nmap:/home/wifi1/Docker/sample-practise# docker network disconnect my_network web-app
```

```
root@nmap:/home/wifi1/Docker/sample-practise# docker network inspect my_network
[
    {
        "Name": "my_network",
        "Id": "75321fe080772c217a6359ba60c845594e22d54bfe1f135774619db9619973af",
        "Created": "2020-05-10T14:50:38.876476756+05:30",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.20.0.0/16",
                    "Gateway": "172.20.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "5eb4f7d06fb89eaf930b76ab80e253b14d2259cf4fb6624caeeff4658fc1e894": {
                "Name": "new-webapp",
                "EndpointID": "9b37e05878d8032150b6e26b90636465c6ce47acd48a5736ae867e5bc2974d11",
                "MacAddress": "02:42:ac:14:00:02",
                "IPv4Address": "172.20.0.2/16",
                "IPv6Address": ""
            },
            "f3e8ba02fe6db158770d14b52f9b4823395129351b3cd79f9b711136c5c83d61": {
                "Name": "web-app",
                "EndpointID": "6b0b0ca5660f61dff90ccef7b936b5ce6a6e50f335382279e0e4e3c13efc5504",
                "MacAddress": "02:42:ac:14:00:03",
                "IPv4Address": "172.20.0.3/16",
                "IPv6Address": ""
            }
        },
        "Options": {},
        "Labels": {}
    }
]
root@nmap:/home/wifi1/Docker/sample-practise#
```



```
root@nmap:/home/wifi1/Docker/sample-practise# docker network --help

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.
root@nmap:/home/wifi1/Docker/sample-practise# docker network connect --help

Usage:  docker network connect [OPTIONS] NETWORK CONTAINER

Connect a container to a network

Options:
      --alias strings           Add network-scoped alias for the container
      --driver-opt strings      driver options for the network
      --ip string               IPv4 address (e.g., 172.30.100.104)
      --ip6 string              IPv6 address (e.g., 2001:db8::33)
      --link list               Add link to another container
      --link-local-ip strings   Add a link-local address for the container
root@nmap:/home/wifi1/Docker/sample-practise# docker network disconnect --help

Usage:  docker network disconnect [OPTIONS] NETWORK CONTAINER

Disconnect a container from a network

Options:
  -f, --force   Force the container to disconnect from a network
root@nmap:/home/wifi1/Docker/sample-practise#
```