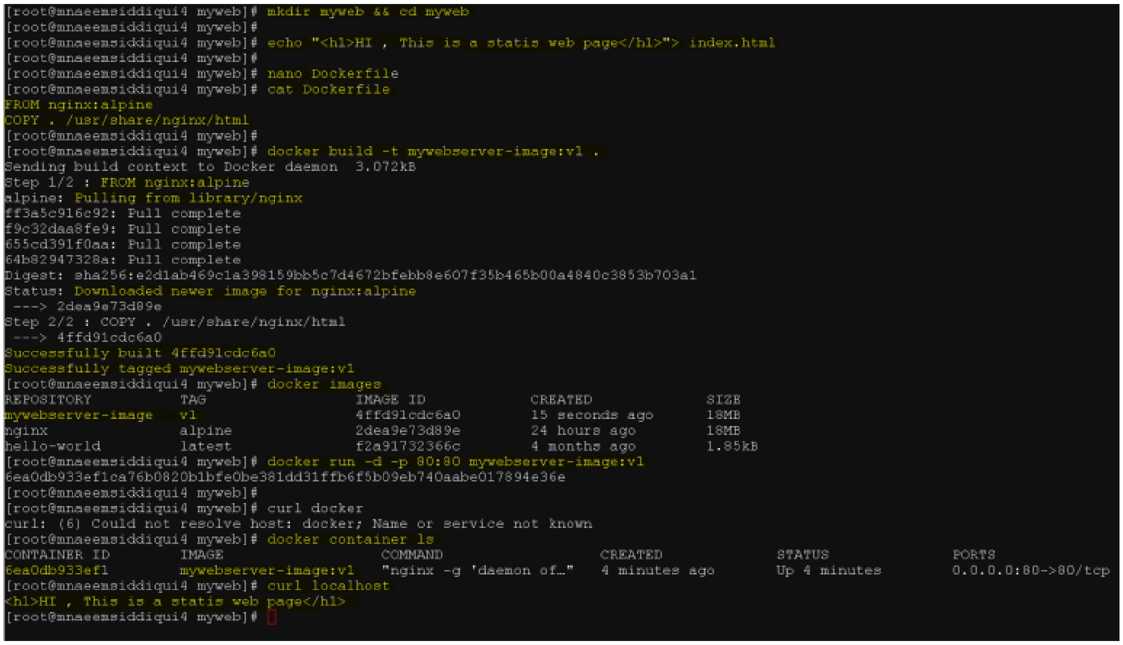
Dockers 101 – Series 6 of N – Using Dockerfile to a static website using nginx server

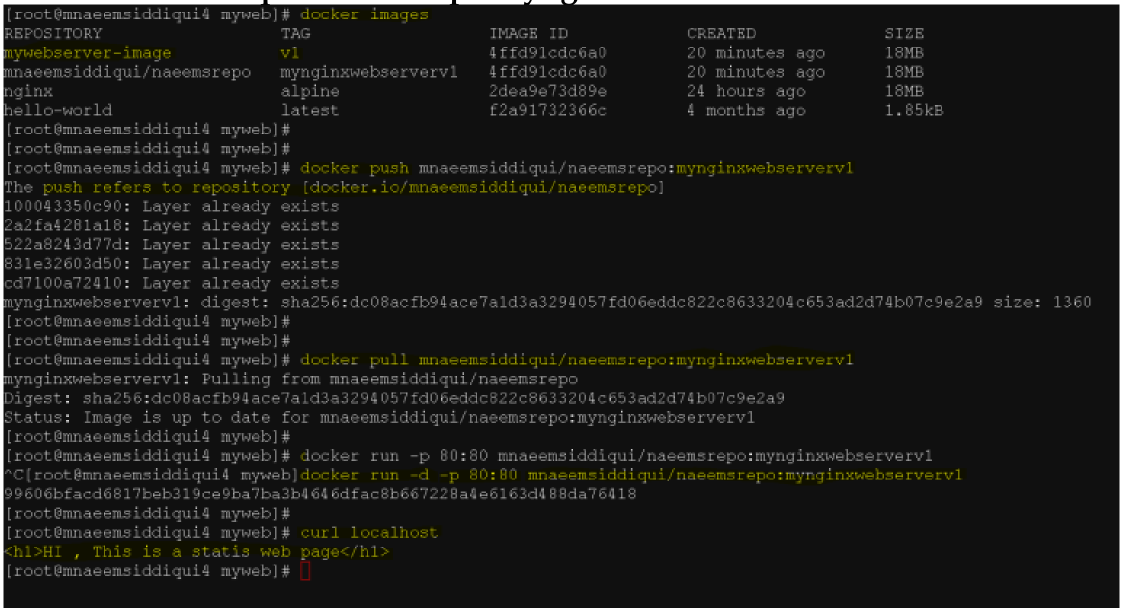
PUBLISHED ON *April 4, 2018April 4, 2018 by Mohd Naeem*

- **Requirement**:
  - To run a static website using nginx server
- **Strategy**:
  - Docker uses a **Dockerfile** to define what all will be going in a container
  - For above requirement we need the following:
    - nginx web server
    - a **working directory** with some static html content
    - copying the contents to nginx server
    - **build** the **app**
    - **push** the **container** to **Docker Hub**(you will need to create Docker Hub account and a repository under the account, Please visit hub.docker.com (http://hub.docker.com))
    - **pull** the **image**
    - **run** the **container**
- **Solution:**
  - Login to your Host machine(in my case a CentOS 7 machine)
  - Make a directory "myweb" and go to the directory – **mkdir myweb && cd myweb**
  - Create a html filr with some content
    - **echo "<h1>HI , This is a statis web page</h1>"> index.html**
  - Now **create a Dockerfile** and copy the following content into it – **nano Dockerfile**
  - **Copy following content** into the **Dockerfile** and **save**:
  - The docker file has self explanatory explanations as what it is doing:
  -
    ```
    FROM nginx:alpine
    COPY . /usr/share/nginx/html
    ```
  -
  - Now build the app-
    - **docker build -t mywebserver-image:v1 .**
  - Now run run the container to run the website
    - **docker run -d -p 80:80 mywebserver-image:v1**
  - Check the content
    - **curl localhost**



```
[root@mnaeemsiddiqui4 myweb]# mkdir myweb && cd myweb
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]# echo "<h1>HI , This is a statis web page</h1>"> index.html
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]# nano Dockerfile
[root@mnaeemsiddiqui4 myweb]# cat Dockerfile
FROM nginx:alpine
COPY . /usr/share/nginx/html
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]# docker build -t mywebserver-image:v1 .
Sending build context to Docker daemon  3.072kB
Step 1/2 : FROM nginx:alpine
alpine: Pulling from library/nginx
ff3a5c916c92: Pull complete
f9c32daa8fe9: Pull complete
655cd391f0aa: Pull complete
64b82947328a: Pull complete
Digest: sha256:e2d1ab469c1a398159bb5c7d4672bfebb8e607f35b465b00a4840c3853b703a1
Status: Downloaded newer image for nginx:alpine
 ---> 2dea9e73d89e
Step 2/2 : COPY . /usr/share/nginx/html
 ---> 4ffd91cdc6a0
Successfully built 4ffd91cdc6a0
Successfully tagged mywebserver-image:v1
[root@mnaeemsiddiqui4 myweb]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
mywebserver-image   v1                  4ffd91cdc6a0        15 seconds ago      18MB
nginx               alpine              2dea9e73d89e        24 hours ago        18MB
hello-world         latest              f2a91732366c        4 months ago        1.85kB
[root@mnaeemsiddiqui4 myweb]# docker run -d -p 80:80 mywebserver-image:v1
6ea0db933ef1ca76b0820b1bfe0be381dd31ffb6f5b09eb740aabe017894e36e
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]# curl docker
curl: (6) Could not resolve host: docker; Name or service not known
[root@mnaeemsiddiqui4 myweb]# docker container ls
CONTAINER ID        IMAGE                  COMMAND                  CREATED             STATUS              PORTS
6ea0db933ef1        mywebserver-image:v1   "nginx -g 'daemon of…"   4 minutes ago       Up 4 minutes        0.0.0.0:80->80/tcp
[root@mnaeemsiddiqui4 myweb]# curl localhost
<h1>HI , This is a statis web page</h1>
[root@mnaeemsiddiqui4 myweb]# 
```

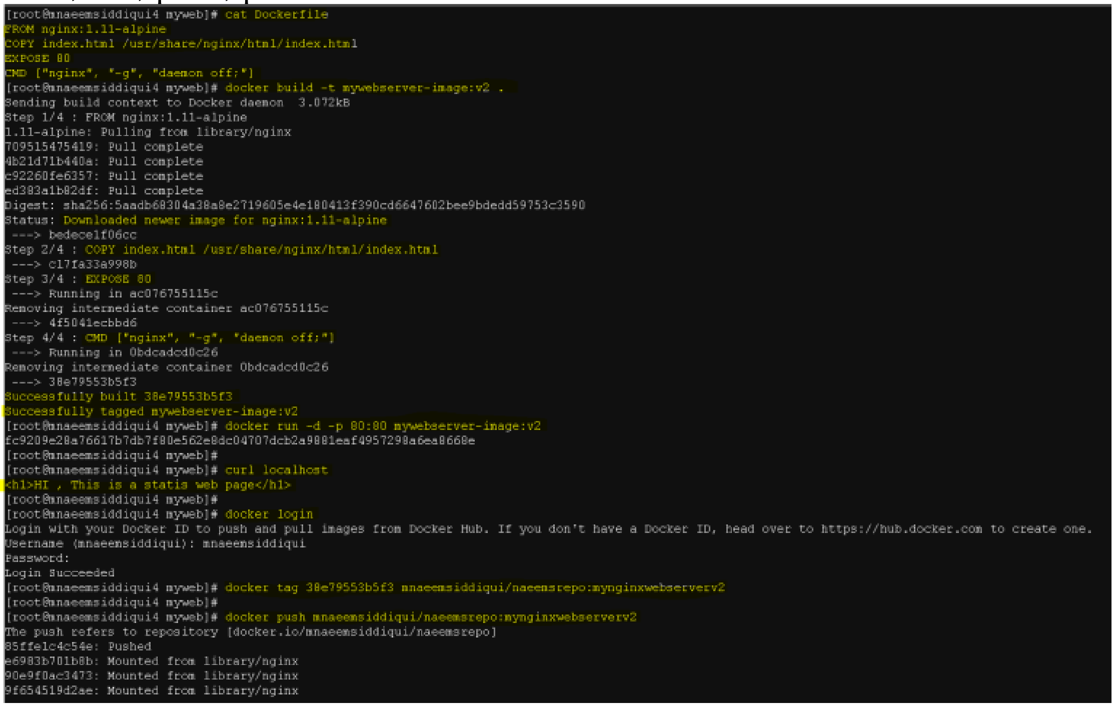
- Now check for the image name for your app and tag it for pushing it to Docker Hub
  - **docker images** # to check for image name
  - **docker tag image username/repository:tag** # for tagging
    - **docker tag 4ffd91cdc6a0 mnaeemsiddiqui/naeemsrepo:mynginxwebserverv1**
  - **docker login** # to login to the Docker hub
- Now push the image to Docker Hub
  - **docker push mnaeemsiddiqui/naeemsrepo:mynginxwebserverv1**
- Now that you have a docker image on docker hub, you can
  - pull the docker image – **docker pull mnaeemsiddiqui/naeemsrepo:mynginxwebserverv1**
  - to run your app – **docker run -d -p 80:80 mnaeemsiddiqui/naeemsrepo:mynginxwebserverv1**



```
[root@mnaeemsiddiqui4 myweb]# docker images
REPOSITORY                  TAG                   IMAGE ID            CREATED             SIZE
mywebserver-image           v1                    4ffd91cdc6a0        20 minutes ago      18MB
mnaeemsiddiqui/naeemsrepo   mynginxwebserverv1    4ffd91cdc6a0        20 minutes ago      18MB
nginx                       alpine                2dea9e73d89e        24 hours ago        18MB
hello-world                 latest                f2a91732366c        4 months ago        1.85kB
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]# docker push mnaeemsiddiqui/naeemsrepo:mynginxwebserverv1
The push refers to repository [docker.io/mnaeemsiddiqui/naeemsrepo]
100043350c90: Layer already exists
2a2fa4281a18: Layer already exists
522a8243d77d: Layer already exists
831e32603d50: Layer already exists
cd7100a72410: Layer already exists
mynginxwebserverv1: digest: sha256:dc08acfb94ace7a1d3a3294057fd06eddc822c8633204c653ad2d74b07c9e2a9 size: 1360
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]# docker pull mnaeemsiddiqui/naeemsrepo:mynginxwebserverv1
mynginxwebserverv1: Pulling from mnaeemsiddiqui/naeemsrepo
Digest: sha256:dc08acfb94ace7a1d3a3294057fd06eddc822c8633204c653ad2d74b07c9e2a9
Status: Image is up to date for mnaeemsiddiqui/naeemsrepo:mynginxwebserverv1
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]# docker run -p 80:80 mnaeemsiddiqui/naeemsrepo:mynginxwebserverv1
^C[root@mnaeemsiddiqui4 myweb]docker run -d -p 80:80 mnaeemsiddiqui/naeemsrepo:mynginxwebserverv1
99606bfacd6817beb319ce9ba7ba3b4646dfac8b667228a4e6163d488da76418
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]# curl localhost
<h1>HI , This is a statis web page</h1>
[root@mnaeemsiddiqui4 myweb]# 
```

- Now update the docker file to add EXPose and CMD commands
- *FROM nginx:1.11-alpine*
  COPY index.html /usr/share/nginx/html/index.html
  EXPOSE 80
  CMD [“nginx”, “-g”, “daemon off;”]

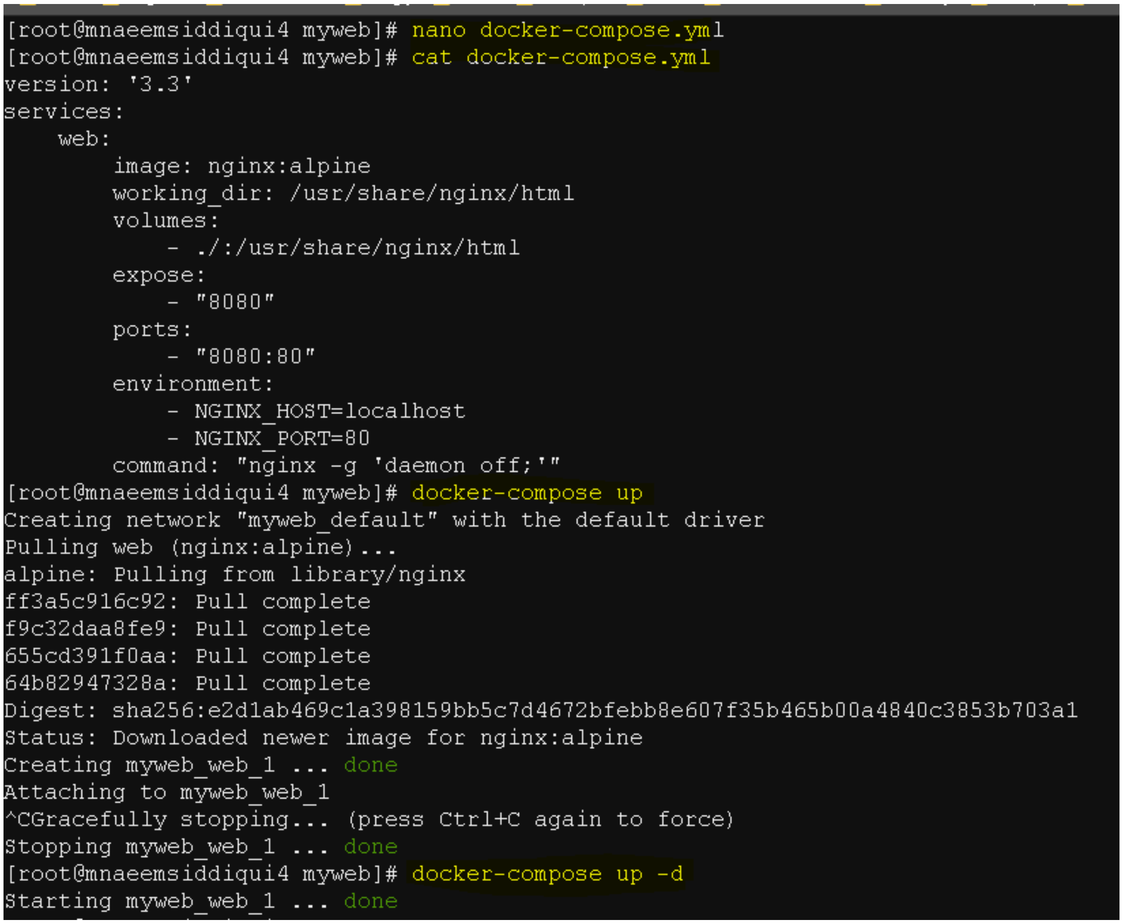
- Build, run, push, pull and run.



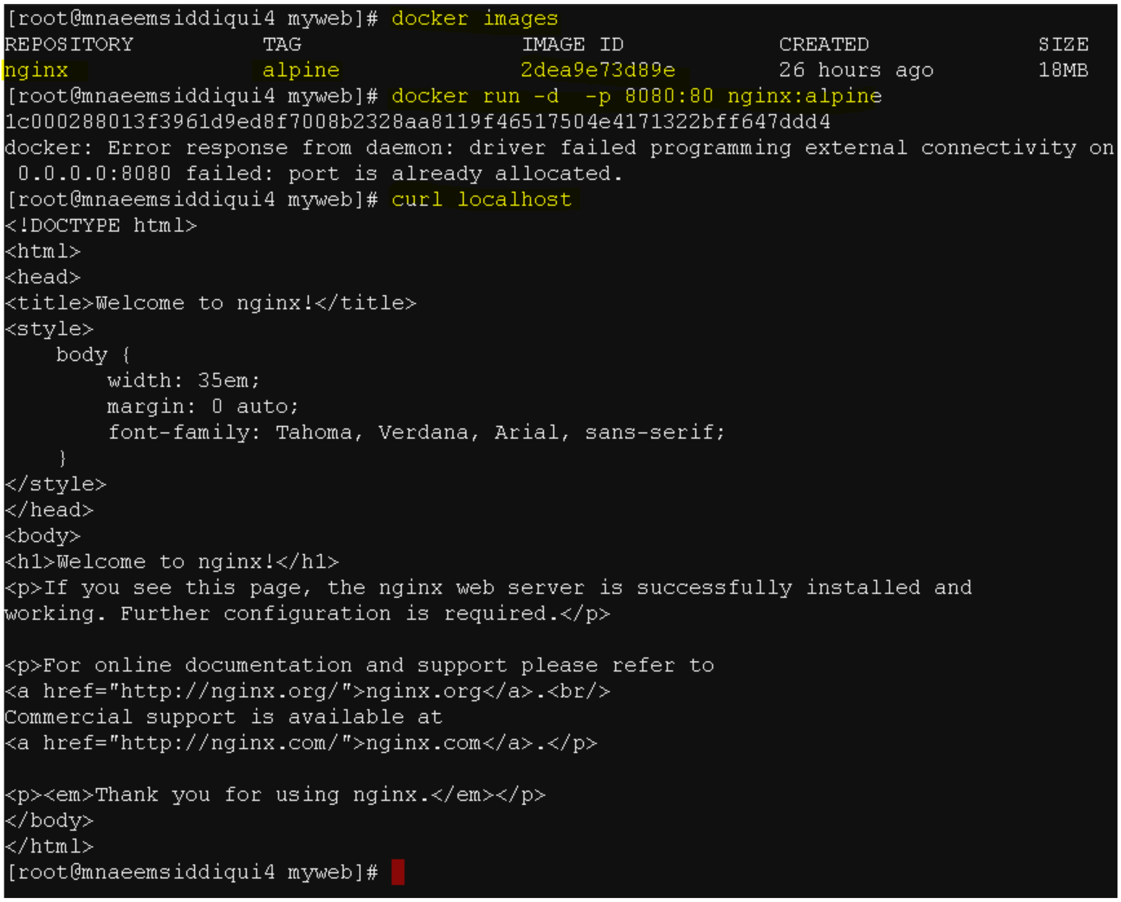
```
[root@mnaeemsiddiqui4 myweb]# cat Dockerfile
FROM nginx:1.11-alpine
COPY index.html /usr/share/nginx/html/index.html
EXPOSE 80
CMD ["nginx", "-g", "daemon off;"]
[root@mnaeemsiddiqui4 myweb]# docker build -t mywebserver-image:v2 .
Sending build context to Docker daemon  3.072kB
Step 1/4 : FROM nginx:1.11-alpine
1.11-alpine: Pulling from library/nginx
709515475419: Pull complete
4b21d71b440a: Pull complete
c92260fe6357: Pull complete
ed383a1b82df: Pull complete
Digest: sha256:5aadb68304a38a8e2719605e4e180413f390cd6647602bee9bdedd59753c3590
Status: Downloaded newer image for nginx:1.11-alpine
 ---> bedece1f06cc
Step 2/4 : COPY index.html /usr/share/nginx/html/index.html
 ---> c17fa33a998b
Step 3/4 : EXPOSE 80
 ---> Running in ac076755115c
Removing intermediate container ac076755115c
 ---> 4f5041ecbbd6
Step 4/4 : CMD ["nginx", "-g", "daemon off;"]
 ---> Running in 0bdcadcd0c26
Removing intermediate container 0bdcadcd0c26
 ---> 38e79553b5f3
Successfully built 38e79553b5f3
Successfully tagged mywebserver-image:v2
[root@mnaeemsiddiqui4 myweb]# docker run -d -p 80:80 mywebserver-image:v2
fc9209e28a76617b7db7f80e562e8dc04707dcb2a9881eaf4957298a6ea8668e
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]# curl localhost
<h1>HI , This is a statis web page</h1>
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]# docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username (mnaeemsiddiqui): mnaeemsiddiqui
Password:
Login Succeeded
[root@mnaeemsiddiqui4 myweb]# docker tag 38e79553b5f3 mnaeemsiddiqui/naeemsrepo:mynginxwebserverv2
[root@mnaeemsiddiqui4 myweb]#
[root@mnaeemsiddiqui4 myweb]# docker push mnaeemsiddiqui/naeemsrepo:mynginxwebserverv2
The push refers to repository [docker.io/mnaeemsiddiqui/naeemsrepo]
85ffe1c4c54e: Pushed
e6983b701b8b: Mounted from library/nginx
90e9f0ac3473: Mounted from library/nginx
9f654519d2ae: Mounted from library/nginx
```

- Now lets use a docker-compose.yml, copy the content below and save.
-
```
version: ‘3.3’
services:

  web:

    image: nginx:alpine
    working_dir: /usr/share/nginx/html
    volumes:
    – ./:/usr/share/nginx/html
    expose:
    – “8080”
    ports:
    – “8080:80”
    environment:
    – NGINX_HOST=localhost
    – NGINX_PORT=80
    command: “nginx -g ‘daemon off;’”
```

- run – **docker compose up -d**

- 

```
[root@mnaeemsiddiqui4 myweb]# nano docker-compose.yml
[root@mnaeemsiddiqui4 myweb]# cat docker-compose.yml
version: '3.3'
services:
    web:
        image: nginx:alpine
        working_dir: /usr/share/nginx/html
        volumes:
            - ./:/usr/share/nginx/html
        expose:
            - "8080"
        ports:
            - "8080:80"
        environment:
            - NGINX_HOST=localhost
            - NGINX_PORT=80
        command: "nginx -g 'daemon off;'"
[root@mnaeemsiddiqui4 myweb]# docker-compose up
Creating network "myweb_default" with the default driver
Pulling web (nginx:alpine)...
alpine: Pulling from library/nginx
ff3a5c916c92: Pull complete
f9c32daa8fe9: Pull complete
655cd391f0aa: Pull complete
64b82947328a: Pull complete
Digest: sha256:e2d1ab469c1a398159bb5c7d4672bfebb8e607f35b465b00a4840c3853b703a1
Status: Downloaded newer image for nginx:alpine
Creating myweb_web_1 ... done
Attaching to myweb_web_1
^CGracefully stopping... (press Ctrl+C again to force)
Stopping myweb_web_1 ... done
[root@mnaeemsiddiqui4 myweb]# docker-compose up -d
Starting myweb_web_1 ... done
```

- 

```
[root@mnaeemsiddiqui4 myweb]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
nginx               alpine              2dea9e73d89e        26 hours ago        18MB
[root@mnaeemsiddiqui4 myweb]# docker run -d  -p 8080:80 nginx:alpine
1c000288013f3961d9ed8f7008b2328aa8119f46517504e4171322bff647ddd4
docker: Error response from daemon: driver failed programming external connectivity on
 0.0.0.0:8080 failed: port is already allocated.
[root@mnaeemsiddiqui4 myweb]# curl localhost
<!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@mnaeemsiddiqui4 myweb]#
```

- Yay!!, you containerized your app and pushed it to docker hub and pulled that image and ran the container to run your application.

CATEGORIES DOCKERS

Powered by WordPress.com.