

Docker Task -3

Task Description:

Create a custom docker image for nginx and deploy it using docker compose, where the volume bind mount should be at /var/opt/nginx location. Push the created custom docker image to your docker-hub.

Techstacks needs to be used :

- AWS EC2
- Docker
- Docker Compose

I choosed a simple To-Do-Application made up using HTML.Then i wrote a docker file to it

```
DO LIST > DOCKERFILE > ...
1  FROM nginx:alpine (last pushed 2 days ago)
2  RUN rm -rf /usr/share/nginx/html/*
3  COPY index.html /usr/share/nginx/html/index.html
4  EXPOSE 80
5  CMD ["nginx", "-g", "daemon off;"]
6
```

Here i choosed to deploy my application through nginx as a reverse proxy Server, where i wrote a docker-file according to it .

I implemented and reduced the size of an image by 95% for faster deployment in the production cycle and cleaned the static content of the nginx to ensure there is no error in the production cycle and application should deploy smoothly and efficiently.

```

● PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker build -t nginx-image .
[+] Building 2.9s (9/9) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 192B
=> [internal] load metadata for docker.io/library/nginx:alpine
=> [auth] library/nginx:pull token for registry-1.docker.io
=> [internal] load .dockignore
=> => transferring context: 2B
=> [1/3] FROM docker.io/library/nginx:alpine@sha256:b3c656d55d7ad751196f21b7fd2e8d4da9cb430e32f646adcf92441b72f82b14
=> => resolve docker.io/library/nginx:alpine@sha256:b3c656d55d7ad751196f21b7fd2e8d4da9cb430e32f646adcf92441b72f82b14
=> [internal] load build context
=> => transferring context: 32B
=> CACHED [2/3] RUN rm -rf /usr/share/nginx/html/*
=> => naming to docker.io/library/nginx-image:latest
=> => unpacking to docker.io/library/nginx-image:latest

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/f9e1915povkqe0q1vxb7cjtox
● PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
nginx-image     latest   31205aa7db74  About an hour ago  79.9MB
● PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker run -d -p 100:80 nginx-image
8a30cbbaf9aed031d9894ce2cf207f0ede4d779679648ecc33454776b23cc
● PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker ps
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
8a30cbbaf9a    nginx-image  "/docker-entrypoint..."  7 seconds ago  Up 6 seconds  0.0.0.0:100->80/tcp, [::]:100->80/tcp  zealous_haslett

=> => exporting manifest list sha256:31205aa7db742a5484c9d2eef7d7e96adc39bcc56afdf9e4c88208c07ec716377
=> => naming to docker.io/library/nginx-image:latest
=> => unpacking to docker.io/library/nginx-image:latest

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/f9e1915povkqe0q1vxb7cjtox
● PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
nginx-image     latest   31205aa7db74  About an hour ago  79.9MB
● PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker run -d -p 100:80 nginx-image
8a30cbbaf9aed031d9894ce2cf207f0ede4d779679648ecc33454776b23cc
● PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker ps
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
8a30cbbaf9a    nginx-image  "/docker-entrypoint..."  7 seconds ago  Up 6 seconds  0.0.0.0:100->80/tcp, [::]:100->80/tcp  zealous_haslett

```

Now the image has been created layer-by-layer, now its time to deploy an image in a cluster / single-node-cluster . Docker-compose will read the docker-file first and create a cluster out of it . Before that we have to tell/ instruct docker-compose to how to do it . For that i am implementing it using yml file to define the services,image,ports and volume to specify my application stored in a cluster.

```

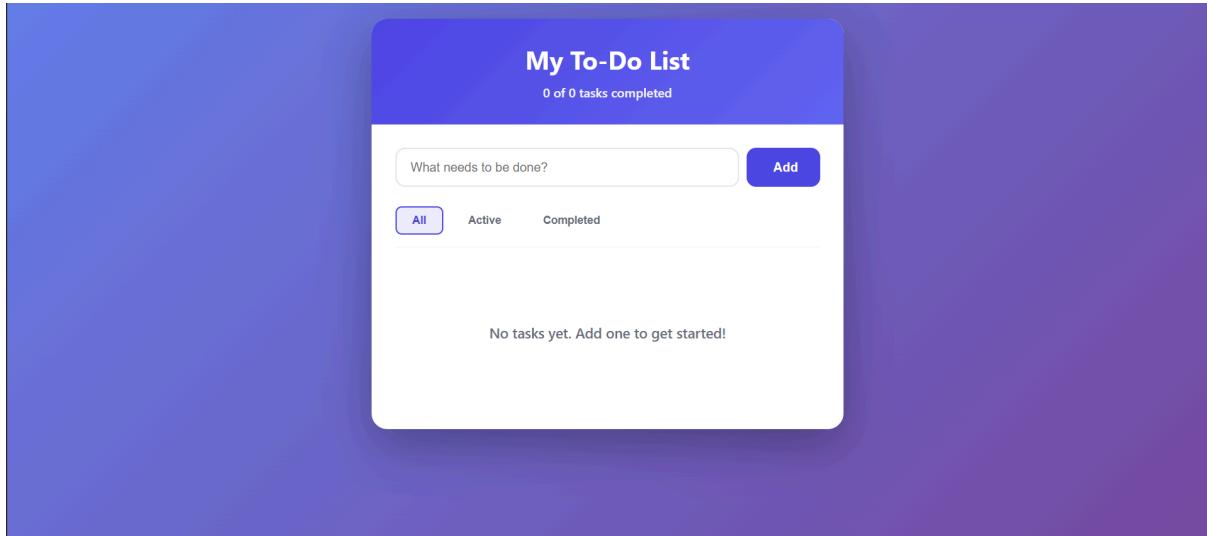
►Run All Services
1 services :
  ►Run Service
2   web:
3     image: nginx:alpine
4     ports:
5       - 200:80
6     volumes:
7       - ./index.html:/usr/share/nginx/html/index.html:ro

```

```

● PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker ps
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
8a30cbbaf9a    nginx-image  "/docker-entrypoint..."  7 seconds ago  Up 6 seconds  0.0.0.0:100->80/tcp, [::]:100->80/tcp  zealous_haslett
● PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker compose ls
NAME      STATUS      CONFIG FILES
● PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker compose up --build -d
[+] Running 1/1
  ✓ web Pulled
[+] Running 1/1
  ✓ Container to-do-list-web-1 Started
● PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List>

```



Now its time to push my image in the docker-hub to make it accessible globally
To do it :

- 1) We have to tag that image first ,to particularly create a repository in a docker hub
- 2) After tagging , push the image to the created repository by specifying the username of the dockerhub

Remember before implementing this step make sure to login your docker in your terminal

```
PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
nginx-image     latest    31205aa7db74  2 hours ago   79.9MB
nginx          alpine    b3c656d55d7a  4 days ago   79.8MB
PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker login
Authenticating with existing credentials... [Username: sheerhsinha]

Info -> To login with a different account, run 'docker logout' followed by 'docker login'

Login Succeeded
PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker tag nginx-image sheerhsinha/dev-test
PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List> docker push sheerhsinha/dev-test
Using default tag: latest
The push refers to repository [docker.io/sheerhsinha/dev-test]
bdabb0d44271: Mounted from sheerhsinha/image-test
ff8a36d5502a: Mounted from sheerhsinha/image-test
72ef17db8f42: Mounted from sheerhsinha/image-test
2d35ebdb57d9: Mounted from sheerhsinha/image-test
df413d6ebdc8: Mounted from sheerhsinha/image-test
194fa24e147d: Mounted from sheerhsinha/image-test
a9978400a692: Mounted from sheerhsinha/image-test
002f62a01bca: Pushed
d9a55dab5954: Mounted from sheerhsinha/image-test
3eaba6cd10a3: Mounted from sheerhsinha/image-test
8f6a6833e95d: Mounted from sheerhsinha/image-test
latest: digest: sha256:31205aa7db742a5484c9d2eef7d7e96adc39bcc56af9e4c88208c07ec716377 size: 856
PS C:\Users\sheer\Documents\docker-task\docker-practice\To-Do-List>
```

