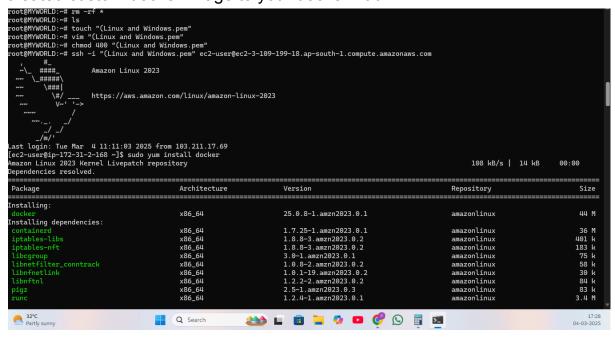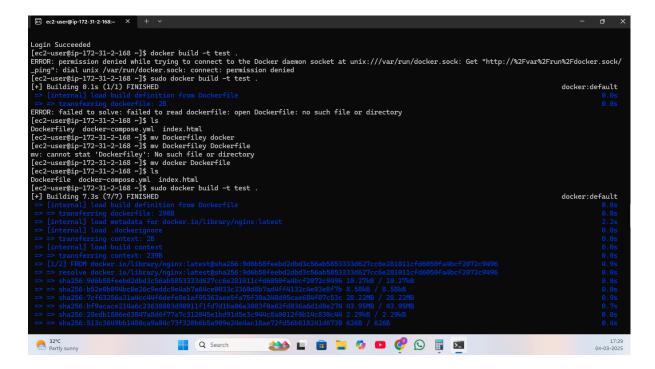Create a custom docker image for nginx and deploy it using docker compose, where the volume bind mount should be at /var/opt/nginx location. Push the created custom docker image to your docker-hub.


```
root@MYWORLD:~# rm -rf *
root@MYWORLD:~# ls
root@MYWORLD:~# touch "(Linux and Windows.pem"
root@MYWORLD:~# vim "(Linux and Windows.pem"
root@MYWORLD:~# chmod 400 "(Linux and Windows.pem"
root@MYWORLD:~# ssh -i "(Linux and Windows.pem" ec2-user@ec2-3-109-199-18.ap-south-1.compute.amazonaws.com
      '   #_
    ~\_  ####_        Amazon Linux 2023
   ~~  \_#####\
   ~~     \###|
   ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
    ~~       V~' '->
     ~~~         /
      ~~._.   _/
         _/ _/
        _/m/'
Last login: Tue Mar  4 11:11:03 2025 from 103.211.17.69
[ec2-user@ip-172-31-2-168 ~]$ sudo yum install docker
Amazon Linux 2023 Kernel Livepatch repository                          108 kB/s |  14 kB     00:00
Dependencies resolved.
================================================================================================
 Package                 Architecture    Version                   Repository          Size
================================================================================================
Installing:
 docker                  x86_64          25.0.8-1.amzn2023.0.1      amazonlinux         44 M
Installing dependencies:
 containerd              x86_64          1.7.25-1.amzn2023.0.1      amazonlinux         36 M
 iptables-libs           x86_64          1.8.8-3.amzn2023.0.2       amazonlinux        401 k
 iptables-nft            x86_64          1.8.8-3.amzn2023.0.2       amazonlinux        183 k
 libcgroup               x86_64          3.0-1.amzn2023.0.1         amazonlinux         75 k
 libnetfilter_conntrack  x86_64          1.0.8-2.amzn2023.0.2       amazonlinux         58 k
 libnfnetlink            x86_64          1.0.1-19.amzn2023.0.2      amazonlinux         30 k
 libnftnl                x86_64          1.2.2-2.amzn2023.0.2       amazonlinux         84 k
 pigz                    x86_64          2.5-1.amzn2023.0.3         amazonlinux         83 k
 runc                    x86_64          1.2.4-1.amzn2023.0.1       amazonlinux        3.4 M
```



```
Login Succeeded
[ec2-user@ip-172-31-2-168 ~]$ docker build -t test .
ERROR: permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/
_ping": dial unix /var/run/docker.sock: connect: permission denied
[ec2-user@ip-172-31-2-168 ~]$ sudo docker build -t test .
[+] Building 0.1s (1/1) FINISHED                                                            docker:default
 => [internal] load build definition from Dockerfile                                               0.0s
 => => transferring dockerfile: 2B                                                                 0.0s
ERROR: failed to solve: failed to read dockerfile: open Dockerfile: no such file or directory
[ec2-user@ip-172-31-2-168 ~]$ ls
Dockerfiley  docker-compose.yml  index.html
[ec2-user@ip-172-31-2-168 ~]$ mv Dockerfiley docker
[ec2-user@ip-172-31-2-168 ~]$ mv Dockerfiley Dockerfile
mv: cannot stat 'Dockerfiley': No such file or directory
[ec2-user@ip-172-31-2-168 ~]$ mv docker Dockerfile
[ec2-user@ip-172-31-2-168 ~]$ ls
Dockerfile  docker-compose.yml  index.html
[ec2-user@ip-172-31-2-168 ~]$ sudo docker build -t test .
[+] Building 7.3s (7/7) FINISHED                                                            docker:default
 => [internal] load build definition from Dockerfile                                               0.0s
 => => transferring dockerfile: 298B                                                               0.0s
 => [internal] load metadata for docker.io/library/nginx:latest                                    2.2s
 => [internal] load .dockerignore                                                                  0.0s
 => => transferring context: 2B                                                                    0.0s
 => [internal] load build context                                                                  0.0s
 => => transferring context: 239B                                                                  0.0s
 => [1/2] FROM docker.io/library/nginx:latest@sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496  4.9s
 => => resolve docker.io/library/nginx:latest@sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496  0.0s
 => => sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496 10.27kB / 10.27kB   0.0s
 => => sha256:b52e0b094bc0e26c9eddc9e4ab7a64ce0033c3360d8b7ad4ff4132c4e03e8f7b 8.58kB / 8.58kB     0.0s
 => => sha256:7cf63256a31a4cc44f6defe8e1af95363aee5fa75f30a248d95cae684f87c53c 28.22MB / 28.22MB   0.9s
 => => sha256:bf9acace214a6c23630803d90911f1fd7d1ba06a3083f0a62fd036a6d1d8e274 43.95MB / 43.95MB   0.7s
 => => sha256:28edb1806e63847a8d6f77a7c312045e1bd91d5e3c944c8a0012f0b14c838c44 2.29kB / 2.29kB     0.0s
 => => sha256:513c3649bb1480ca9a04c73f320b6b5a909e24e4ac18ae72fd56b818241d6730 626B / 626B         0.4s
```

```
55e9644f21c3: Preparing
7d22e2347c12: Preparing
f6d5815f290e: Preparing
791f0a07985c: Preparing
cabea05c000e: Waiting
c68632c455ae: Waiting
5f1ee22ffb5e: Waiting
denied: requested access to the resource is denied
[ec2-user@ip-172-31-2-168 ~]$ sudo docker images
REPOSITORY                TAG        IMAGE ID       CREATED         SIZE
test                      latest     826c08efcaa8   3 minutes ago   192MB
pallavidocker2024/nginx   latest     826c08efcaa8   3 minutes ago   192MB
[ec2-user@ip-172-31-2-168 ~]$ sudo docker login
Log in with your Docker ID or email address 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.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizatio
ns using SSO. Learn more at https://docs.docker.com/go/access-tokens/

Username: pallavidocker2024
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[ec2-user@ip-172-31-2-168 ~]$ sudo docker push  pallavidocker2024/nginx
Using default tag: latest
The push refers to repository [docker.io/pallavidocker2024/nginx]
f2ca847749bb3: Pushed
55e9644f21c3: Mounted from library/nginx
7d22e2347c12: Mounted from library/nginx
f6d5815f290e: Mounted from library/nginx
791f0a07985c: Mounted from library/nginx
cabea05c000e: Mounted from library/nginx
c68632c455ae: Mounted from library/nginx
```

```
Last login: Tue Mar  4 11:23:32 2025 from 103.211.17.69
[ec2-user@ip-172-31-2-168 ~]$ sudo docker exec -it ^C
[ec2-user@ip-172-31-2-168 ~]$ sudo docker exec -it dcf0b25d876d
"docker exec" requires at least 2 arguments.
See 'docker exec --help'.

Usage:  docker exec [OPTIONS] CONTAINER COMMAND [ARG...]

Execute a command in a running container
[ec2-user@ip-172-31-2-168 ~]$ sudo docker exec -it dcf0b25d876d sh
# cd /usr/share/nginx/html
# ls
# exit
[ec2-user@ip-172-31-2-168 ~]$  cd /usr/share/nginx/html
-bash: cd: /usr/share/nginx/html: No such file or directory
[ec2-user@ip-172-31-2-168 ~]$ sudo docker pull pallavidocker2024/nginx:latest
latest: Pulling from pallavidocker2024/nginx
Digest: sha256:e3ab554d70773f908d2cf6b6c4f06f87d8a31ab9e5379eff2660bb2542d703f6
Status: Image is up to date for pallavidocker2024/nginx:latest
docker.io/pallavidocker2024/nginx:latest
[ec2-user@ip-172-31-2-168 ~]$ sudo docker images
REPOSITORY                TAG        IMAGE ID       CREATED          SIZE
test                      latest     826c08efcaa8   16 minutes ago   192MB
pallavidocker2024/nginx   latest     826c08efcaa8   16 minutes ago   192MB
[ec2-user@ip-172-31-2-168 ~]$ sudo docker run -itd -p 80:80
"docker run" requires at least 1 argument.
See 'docker run --help'.

Usage:  docker run [OPTIONS] IMAGE [COMMAND] [ARG...]

Create and run a new container from an image
[ec2-user@ip-172-31-2-168 ~]$ sudo docker run -itd -p 80:80 826c08efcaa8
57fdc32000202012fabef7d211a374fa6635ac77072d31fec7b74eb3336ba79c
[ec2-user@ip-172-31-2-168 ~]$
```

app.docker.com/settings/personal-access-tokens/copy?description=nginx&scope=repo%3Aadmin&expiration=

Password

2FA

Personal access tokens

Connected accounts

Convert

Deactivate

Copy access token

Use this token as a password when you sign in from the Docker CLI client. Learn more ⬚

Make sure you copy your personal access token now. Your personal access token is only displayed once. It isn't stored and can't be retrieved later.

Access token description
nginx

Expires on
Never

Access permissions
Read, Write, Delete

To use the access token from your Docker CLI client:

1. Run

```
$  docker login -u pallavidocker2024                                    Copy
```

2. At the password prompt, enter the personal access token.

```
dckr_pat_6NwaS-8Oxu3IYYtdQG-aAM5K2f4                                    Copy
```

Back to access tokens

32°C
Partly sunny
Q Search
17:30
04-03-2025

---

pallavidocker2024 / Repositories / nginx / Tags

Using 0 of 1 private repositories.

pallavidocker2024/nginx ⬚

Last pushed 4 minutes ago · Repository size: 68.8 MB

Docker commands   Public view

To push a new tag to this repository:

```
docker push pallavidocker2024/nginx:tagname
```

Add a description ✎ ⓘ

Add a category ✎ ⓘ

General   **Tags**   Image Management BETA   Builds   Collaborators   Webhooks   Settings

| | Sort by | Newest ⌄ | 🔍 Filter tags | Delete |

TAG

● latest

Last pushed 9 minutes by pallavidocker2024

```
docker pull pallavidocker2024/nginx:latest    Copy
```

Digest	OS/ARCH	Last pull	Compressed size ⓘ
e3ab554d7077	linux/amd64	less than 1 day	68.83 MB

32°C
Partly sunny
Q Search
17:30
04-03-2025

Welcome to My Custom Nginx!

Welcome to My Custom Nginx!