

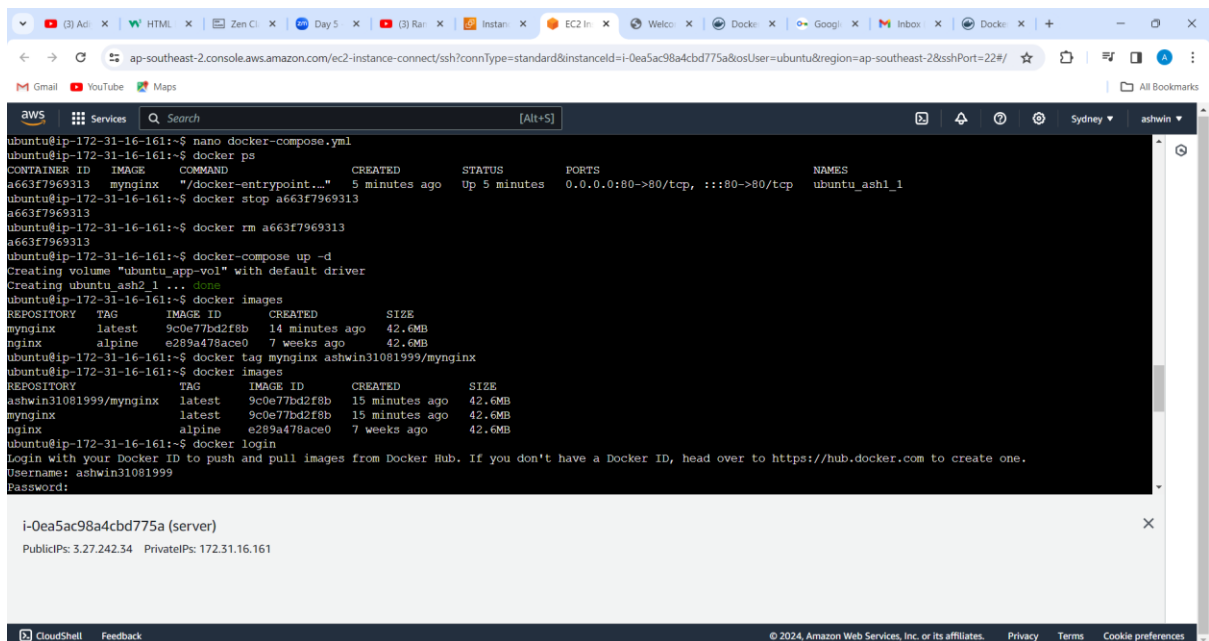
The screenshot shows a terminal window in AWS CloudShell. The user is on an Ubuntu instance and has run the following commands: `mkdir nginx`, `nano index.html`, `nano Dockerfile`, and `docker build -t mynginx .`. The build process is shown, including pulling the nginx:alpine image and building the Docker image. The output shows the image is successfully built and pushed to Docker Hub. The terminal also shows the Dockerfile content: `FROM nginx:alpine`, `WORKDIR /usr/share/nginx/html`, and `COPY index.html /usr/share/nginx/html`. The final output shows the image is successfully built and pushed to Docker Hub.

```
ubuntu@ip-172-31-16-161:~$ mkdir nginx
ubuntu@ip-172-31-16-161:~$ nano index.html
ubuntu@ip-172-31-16-161:~$ nano Dockerfile
ubuntu@ip-172-31-16-161:~$ docker build -t mynginx .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/

Sending build context to Docker daemon 16.38kB
Step 1/2 : FROM nginx:alpine
alpine: Pulling from library/nginx
619be1103602: Pull complete
ed9e62e73b3b: Pull complete
6126dca06df7: Pull complete
1d0dd42dc2265: Pull complete
2blab92f0231: Pull complete
6eba808ac059: Pull complete
57038e85fbb8: Pull complete
eec94c9845c0: Pull complete
Digest: sha256:31bad00311cb5eeb8a6648beadcf67277a175da89989f14727420a80e2e76742
Status: Downloaded newer image for nginx:alpine
--> e289a478ace0
Step 2/2 : COPY index.html /usr/share/nginx/html/
--> 9c0e77bd2f8b
Successfully built 9c0e77bd2f8b

i-Oea5ac98a4cbd775a (server)
PublicIPs: 3.27.242.34 PrivateIPs: 172.31.16.161
```

1 (creating nginx custom image and push to docker hub)



The screenshot shows a terminal window in AWS CloudShell. The user is on an Ubuntu instance and has run the following commands: `nano docker-compose.yml`, `docker ps`, `docker stop a663f7969313`, `docker rm a663f7969313`, `docker compose up -d`, and `docker images`. The output shows the Docker compose up command creating a volume and starting the container. The terminal also shows the Dockerfile content: `FROM nginx:alpine`, `WORKDIR /usr/share/nginx/html`, and `COPY index.html /usr/share/nginx/html`. The final output shows the image is successfully built and pushed to Docker Hub.

```
ubuntu@ip-172-31-16-161:~$ nano docker-compose.yml
ubuntu@ip-172-31-16-161:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
a663f7969313   mynginx   "/docker-entrypoint..." 5 minutes ago  Up 5 minutes  0.0.0.0:80->80/tcp, :::80->80/tcp  ubuntu_ash1_1
ubuntu@ip-172-31-16-161:~$ docker stop a663f7969313
a663f7969313
ubuntu@ip-172-31-16-161:~$ docker rm a663f7969313
a663f7969313
ubuntu@ip-172-31-16-161:~$ docker compose up -d
Creating volume "ubuntu_app-vol" with default driver
Creating ubuntu_ash2_1 ... done
ubuntu@ip-172-31-16-161:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
mynginx       latest   9c0e77bd2f8b   14 minutes ago  42.6MB
nginx         alpine   e289a478ace0   7 weeks ago    42.6MB
ubuntu@ip-172-31-16-161:~$ docker tag mynginx ashwin31081999/mynginx
ubuntu@ip-172-31-16-161:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
ashwin31081999/mynginx latest   9c0e77bd2f8b   15 minutes ago  42.6MB
mynginx       latest   9c0e77bd2f8b   15 minutes ago  42.6MB
nginx         alpine   e289a478ace0   7 weeks ago    42.6MB
ubuntu@ip-172-31-16-161:~$ 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: ashwin31081999
Password:
```

2 (creating nginx custom image and push to docker hub)

The screenshot shows a terminal window in AWS CloudShell. The user is logged in as 'ashwin31081999'. They attempt to login to Docker Hub with the username 'ashwin31081999' and a password. The login is successful. Then, they push a Docker image named 'mynginx' to Docker Hub. The push is successful, and the image is available on Docker Hub. The terminal output shows the following commands and their results:

```
ashwin31081999@ubuntu:~$ docker login
Error response from daemon: Get "https://registry-1.docker.io/v2/": unauthorized: incorrect username or password
ashwin31081999@ubuntu:~$ 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: ashwin31081999
Password:
WARNING! Your password will be stored unencrypted in /home/ubuntu/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
ashwin31081999@ubuntu:~$ docker push ashwin31081999/mynginx
Using default tag: latest
The push refers to repository [docker.io/ashwin31081999/mynginx]
3d8c2185dbfa: Pushed
13c52683b537: Mounted from library/nginx
337b7d64083b: Mounted from library/nginx
odd311f34c29: Mounted from library/nginx
3e8ad8bcb0ac: Mounted from library/nginx
74b4ff8dbbd1: Mounted from library/nginx
c018a48a857c: Mounted from library/nginx
0f73163669d4: Mounted from library/nginx
aadc3bda2944: Mounted from library/nginx
latest: digest: sha256:5176fa2a5951e28ccf9f7a7477916ac3cebedfe8d565def3bfec301a3f56a1da size: 2196
```

3 (creating nginx custom image and push to docker hub)

The screenshot shows a terminal window in AWS CloudShell. The user is logged in as 'ashwin31081999'. They run the command 'docker ps' to list the running containers. The output shows a single container named 'ubuntu_ash2_1' with the image 'mynginx' and the command '/docker-entrypoint...'. The container is created 21 minutes ago and is up for 21 minutes. The ports are 0.0.0.0:80->80/tcp and :::80->80/tcp.

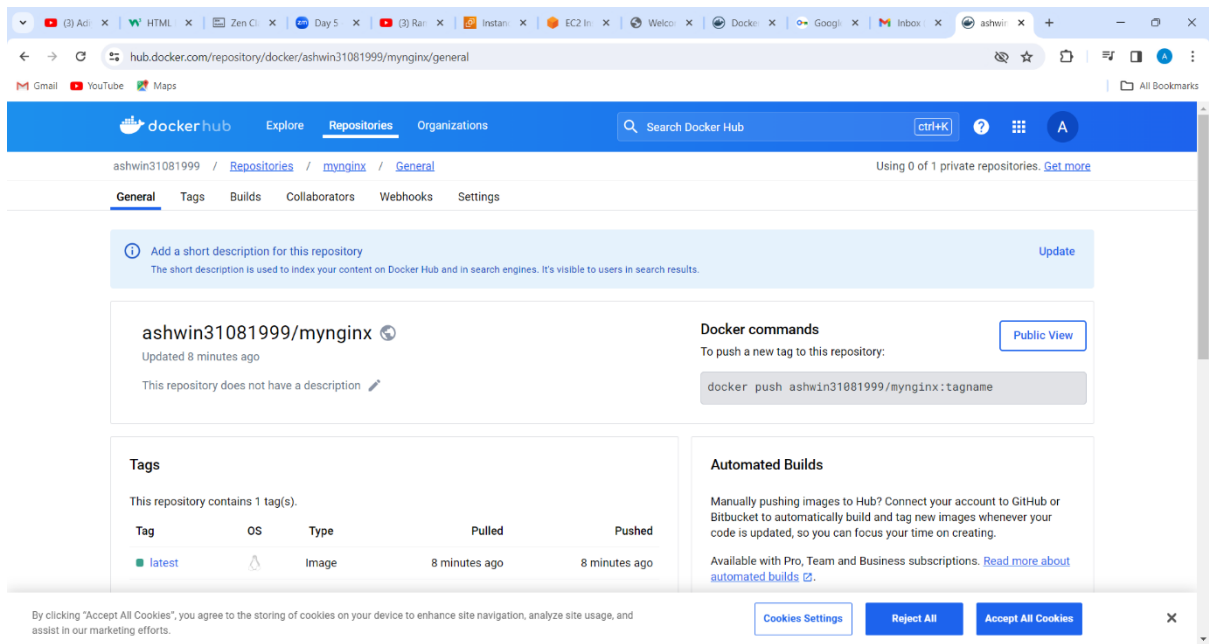
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
0e395d7ae484	mynginx	"/docker-entrypoint..."	21 minutes ago	Up 21 minutes	0.0.0.0:80->80/tcp, :::80->80/tcp	ubuntu_ash2_1

4 (creating nginx custom image and push to docker hub)

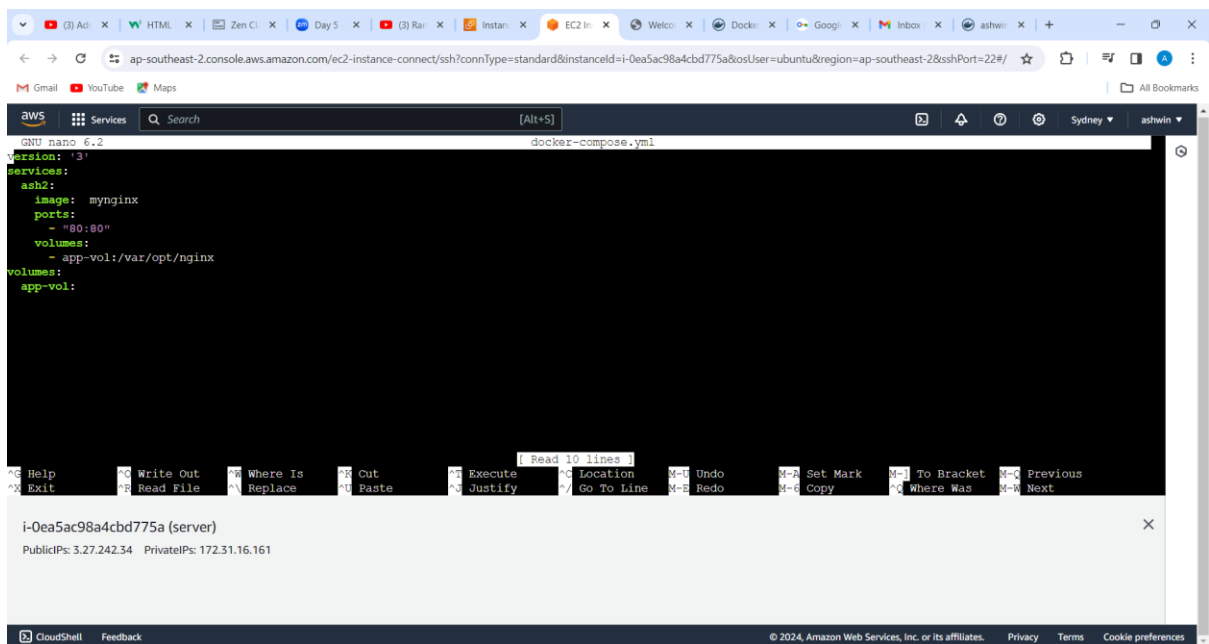
The screenshot shows a terminal window in AWS CloudShell. The user is logged in as 'ashwin31081999'. They are in the directory '/usr/share/nginx/html/' and are creating a Dockerfile. The Dockerfile content is as follows:

```
FROM nginx:alpine
COPY index.html /usr/share/nginx/html/
```

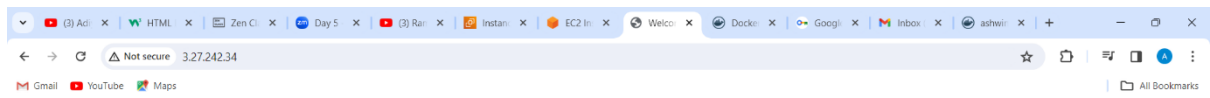
5 (Dockerfile)



6(docker hub)



7(docker-compose.yml)



Welcome to My Nginx Server!

If you see this page, the Nginx web server is successfully installed and working.

task 9(output)