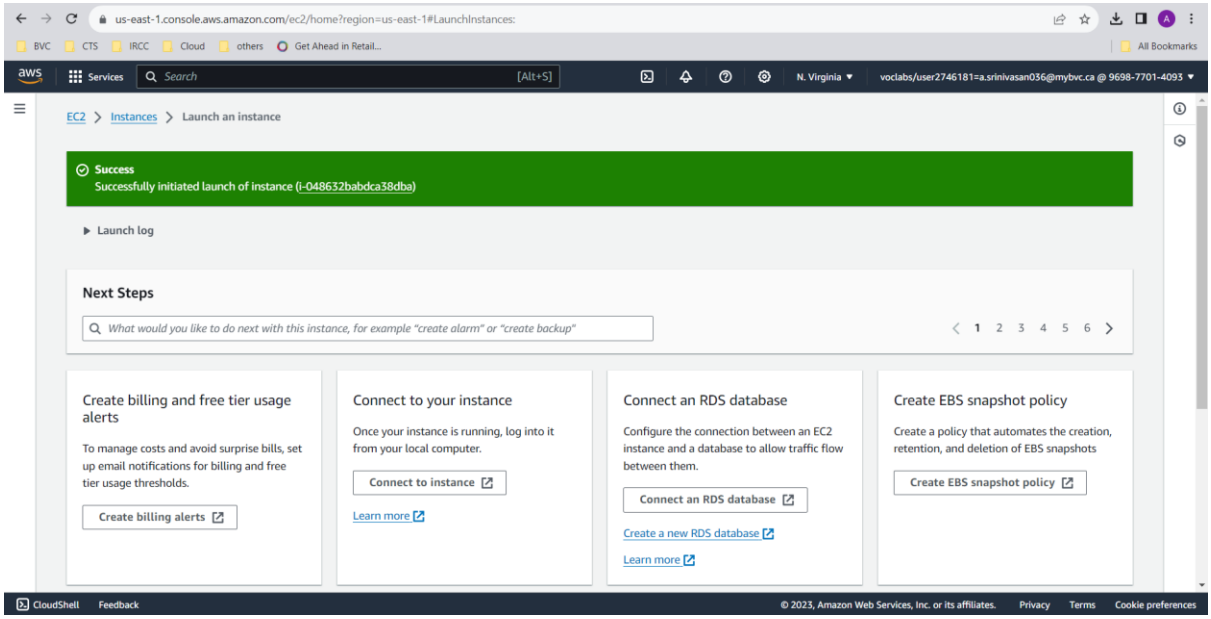
Akilandeshwari Srinivasan(451036)

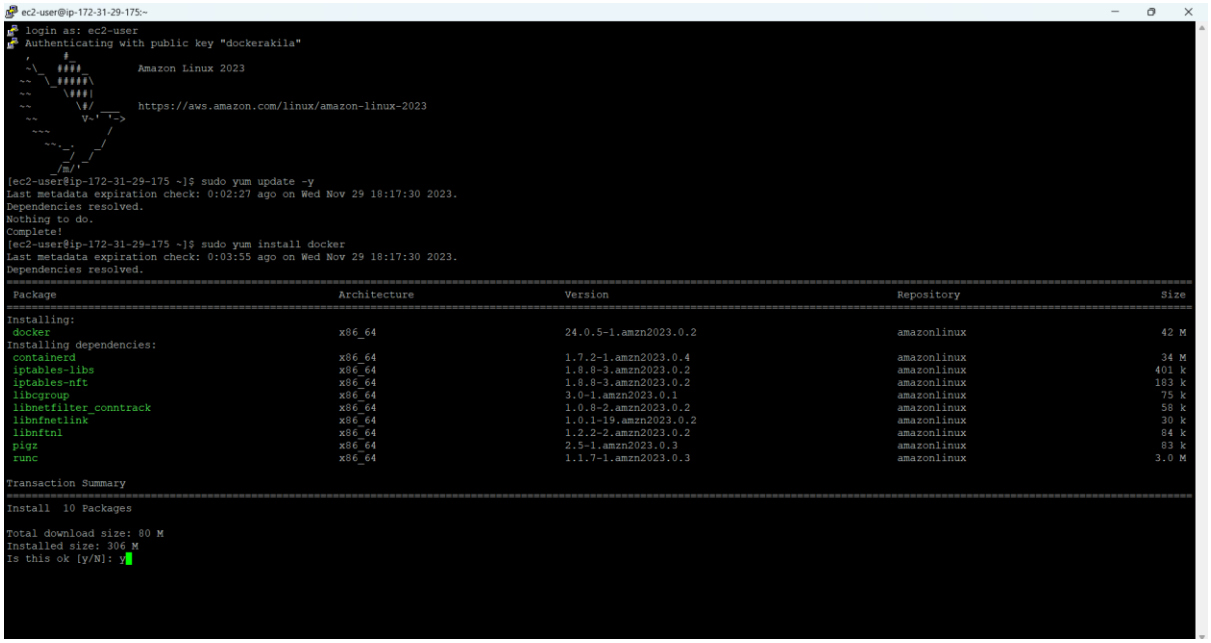
In-class Exercise _ CLCM3504 ---Deploying on Ec2 using Docker

1. Create ec2 instance and connect with docker.

Create an ec2 instance



Connect to the putty and update & install docker with the cmds " sudo yum update -y" & " sudo yum install docker"



Start the docker

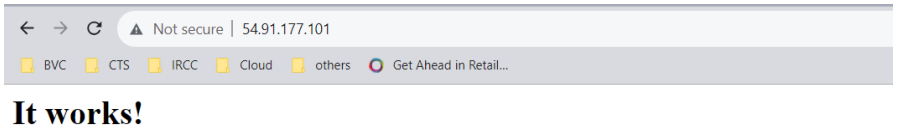
```
Complete!
[ec2-user@ip-172-31-29-175 ~]$ sudo service docker start
Redirecting to /bin/systemctl start docker.service
[ec2-user@ip-172-31-29-175 ~]$
```

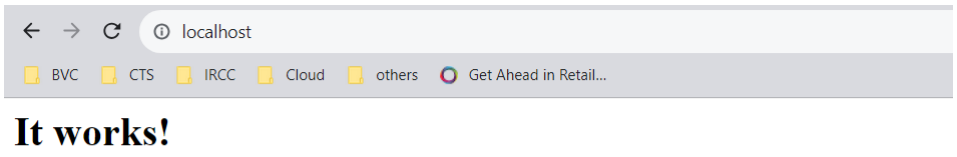
Run the docker using the same command

```
  GitCommit:        de40ad0
[ec2-user@ip-172-31-29-175 ~]$ docker run -d -p 80:80 httpd
0042f54c509f680187ec69a601124fc4cc65644f03eb2fa6888a70156968746e
[ec2-user@ip-172-31-29-175 ~]$
```

```
[ec2-user@ip-172-31-29-175 ~]$ docker ps
CONTAINER ID   IMAGE     COMMAND              CREATED          STATUS          PORTS                                   NAMES
0042f54c509f   httpd     "httpd-foreground"   24 seconds ago   Up 22 seconds   0.0.0.0:80->80/tcp, :::80->80/tcp   bold_sammet
[ec2-user@ip-172-31-29-175 ~]$
```

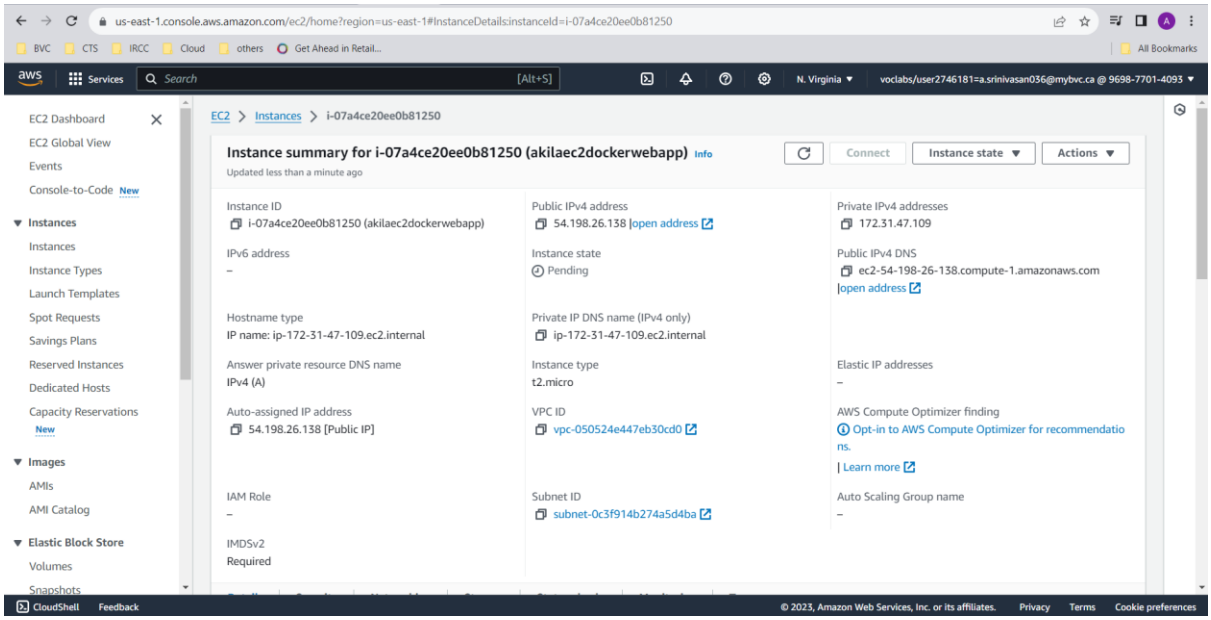
You can see the output by navigating to the public DNS of the EC2 instance





2. hosting webapp in ec2 with docker

Create an another EC2 instance.



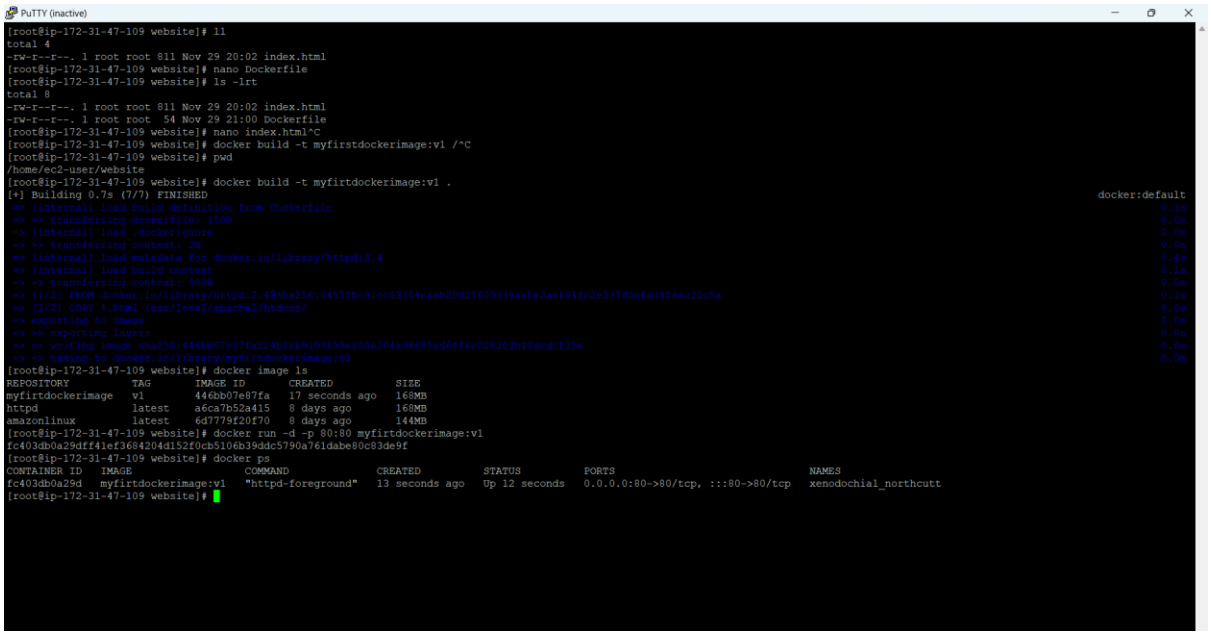
Connect to the putty and install the docker like the one which done before.

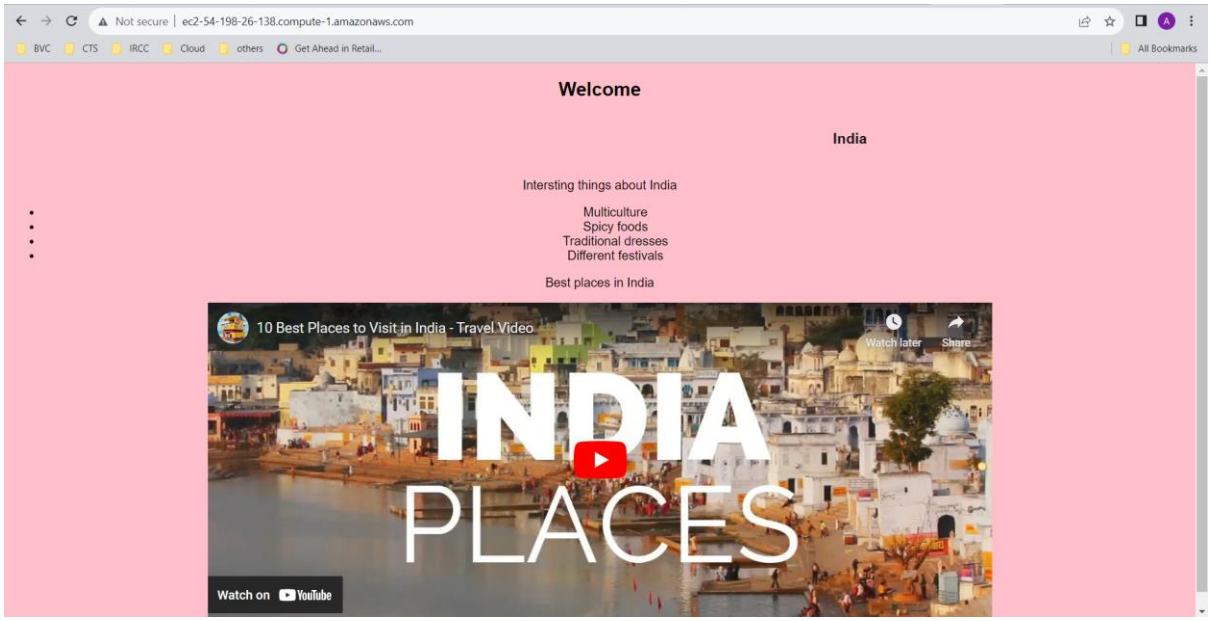
Now here we have to create a docker file and put the commands instead of taking pull and manually moving html file to the server.

FROM httpd:2.4 -----→ pull apache

COPY ./public-html/ /usr/local/apache2/htdocs/ ----------→ copy the file inside the folder (public-html) to the other path.

Then build the Dockerfile and run it.


PuTTY (inactive)
[root@ip-172-31-47-109 website]# ll
total 4
-rw-r--r--. 1 root root 811 Nov 29 20:02 index.html
[root@ip-172-31-47-109 website]# nano Dockerfile
[root@ip-172-31-47-109 website]# ls -lrt
total 8
-rw-r--r--. 1 root root 811 Nov 29 20:02 index.html
-rw-r--r--. 1 root root 54 Nov 29 21:00 Dockerfile
[root@ip-172-31-47-109 website]# nano index.html^C
[root@ip-172-31-47-109 website]# docker build -t myfirstdockerimage:v1 /^C
[root@ip-172-31-47-109 website]# pwd
/home/ec2-user/website
[root@ip-172-31-47-109 website]# docker build -t myfirtdockerimage:v1 .
[+] Building 0.7s (7/7) FINISHED
docker:default
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 150B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/httpd:2.4
=> [internal] load build context
=> => transferring context: 909B
=> [1/2] FROM docker.io/library/httpd:2.4@sha256:04551b91cc03314eaab20d23609339aebe2ae694fc2e337d0afad429ec22c5a
=> [2/2] COPY *.html /usr/local/apache2/htdocs/
=> exporting to image
=> => exporting layers
=> => writing image sha256:446bb07e87fa224bfeb9109699e830e204a94685ad6ff6c026303bffdcdef35e
=> => naming to docker.io/library/myfirtdockerimage:v1
[root@ip-172-31-47-109 website]# docker image ls
REPOSITORY TAG IMAGE ID CREATED SIZE
myfirtdockerimage v1 446bb07e87fa 17 seconds ago 168MB
httpd latest a6ca7b52a415 8 days ago 168MB
amazonlinux latest 6d7779f20f70 8 days ago 144MB
[root@ip-172-31-47-109 website]# docker run -d -p 80:80 myfirtdockerimage:v1
fc403db0a29dff41ef3684204d152f0cb5106b39ddc5790a761dabe80c83de9f
[root@ip-172-31-47-109 website]# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
fc403db0a29d myfirtdockerimage:v1 "httpd-foreground" 13 seconds ago Up 12 seconds 0.0.0.0:80->80/tcp, :::80->80/tcp xenodochial_northcutt
[root@ip-172-31-47-109 website]#


Not secure | ec2-54-198-26-138.compute-1.amazonaws.com
BVC
CTS
IRCC
Cloud
others
Get Ahead in Retail...
All Bookmarks
Welcome
India
Intersting things about India
Multiculture
Spicy foods
Traditional dresses
Different festivals
Best places in India
10 Best Places to Visit in India - Travel Video
Watch later
Share
INDIA
PLACES
Watch on YouTube