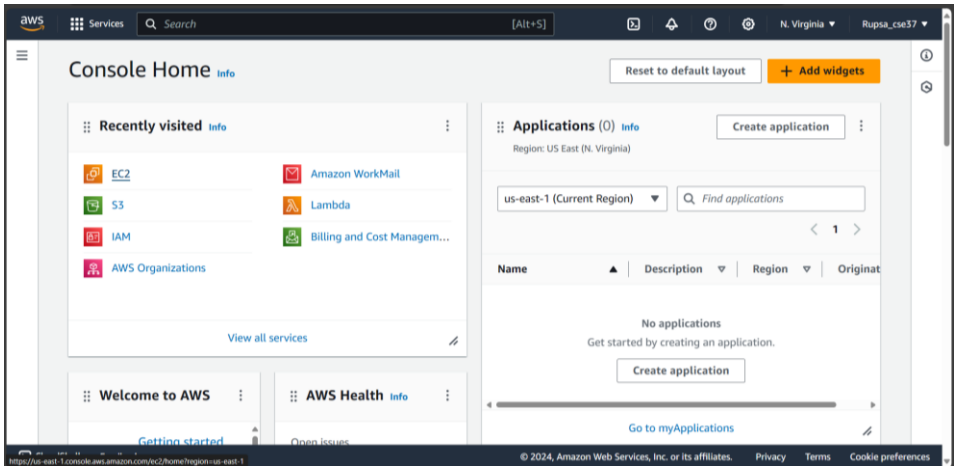
PROBLEM STATEMENT :

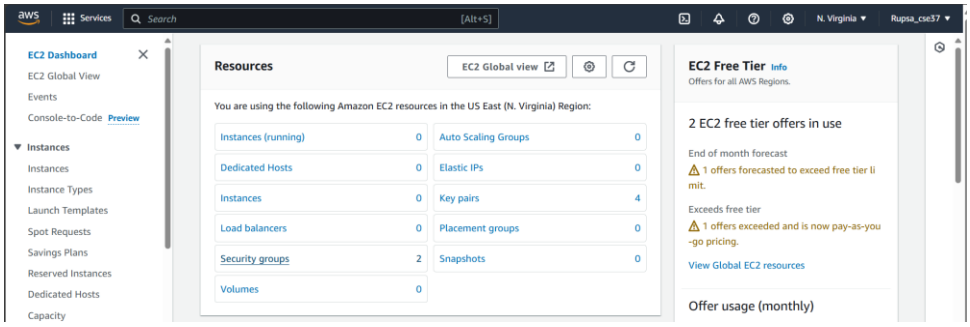
10) Deploy and run the project in AWS without using the port.

Steps to deploy and run the project without using the port:

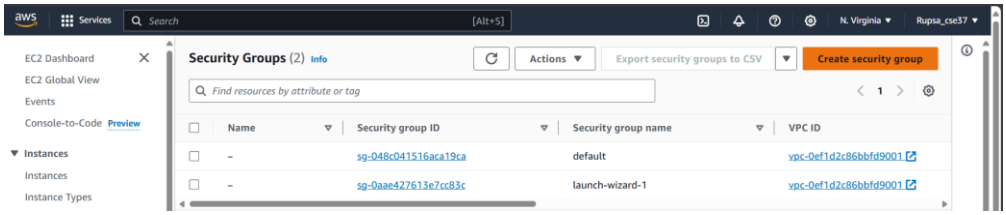
1. Sign up for an AWS account, search for ‘EC2’ then click on it.



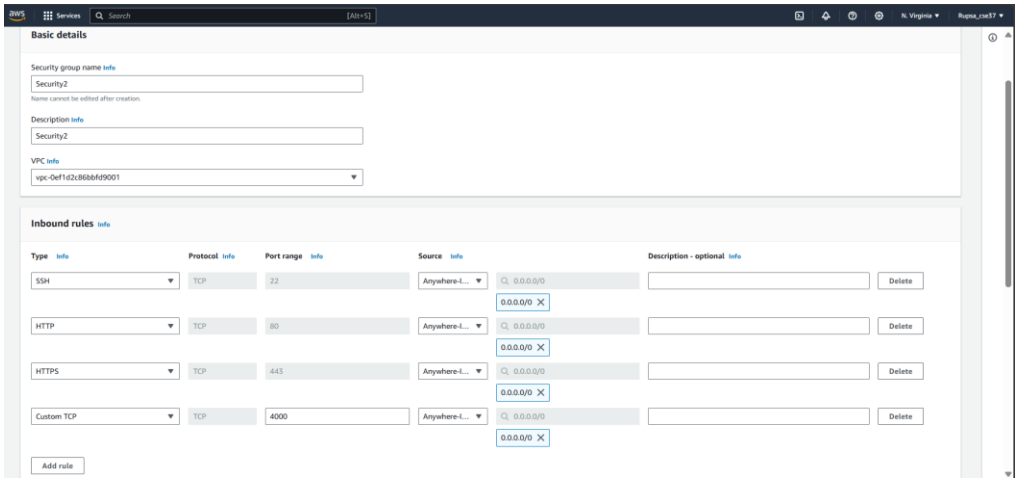
2. Now click on security group.



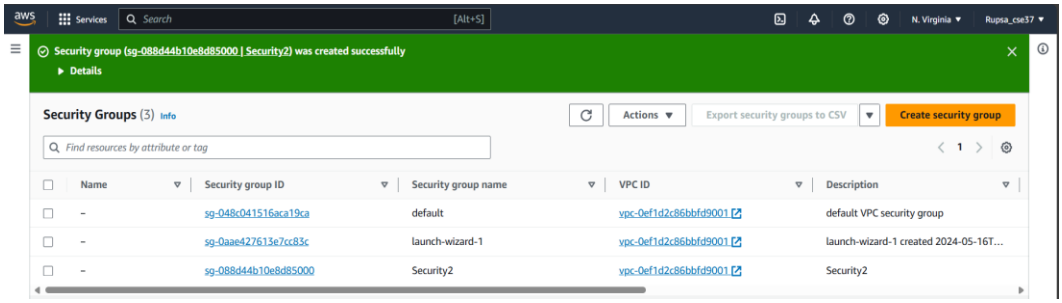
3. Click on “Create security group”.



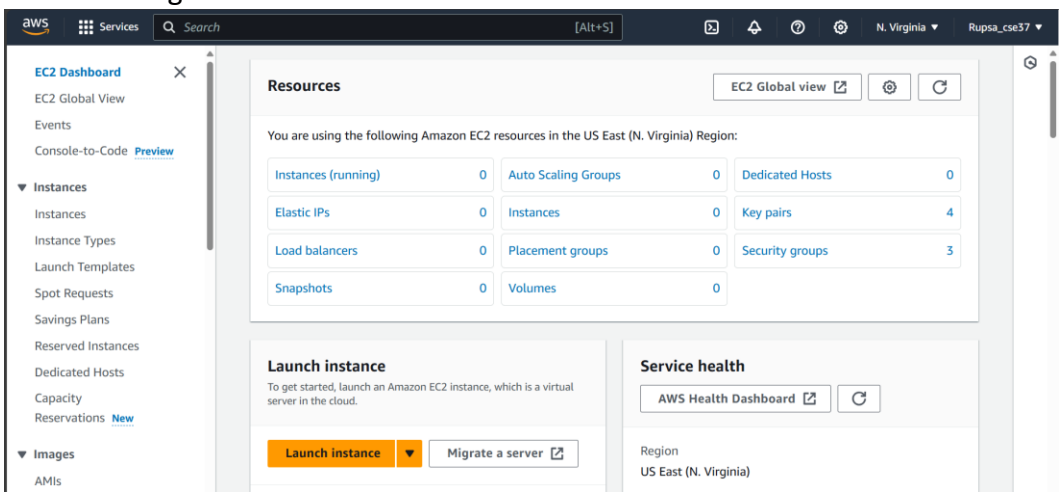
4. Give the name then in “Inbound rules” click on “Add rule” & add SSH,HTTP,HTTPS,CUSTOM TCP protocol and then click on “Create security group”.



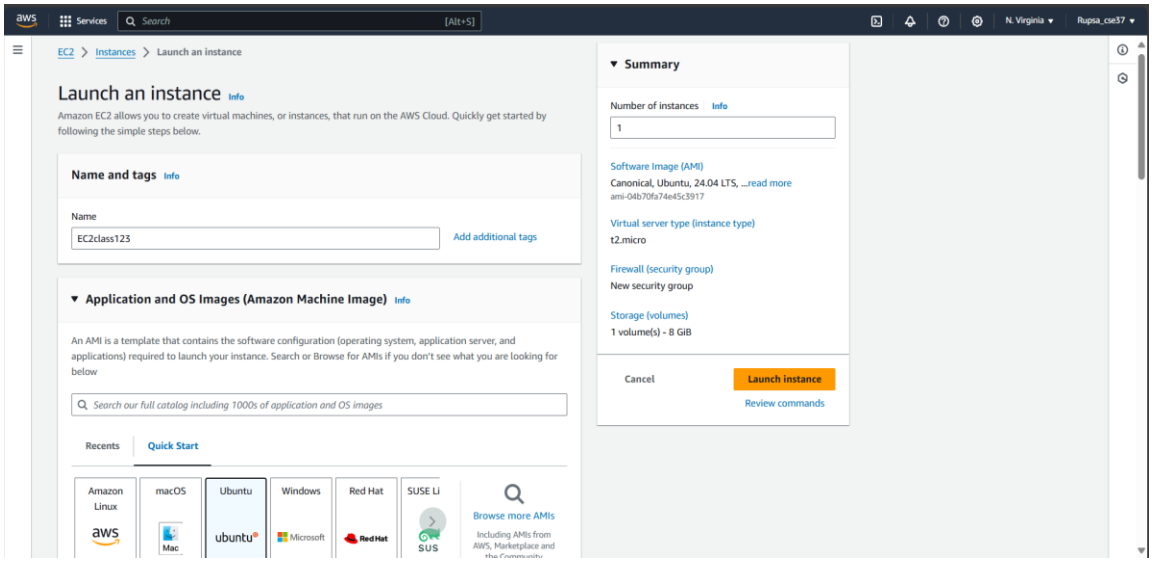
5. Security2 is successfully created.

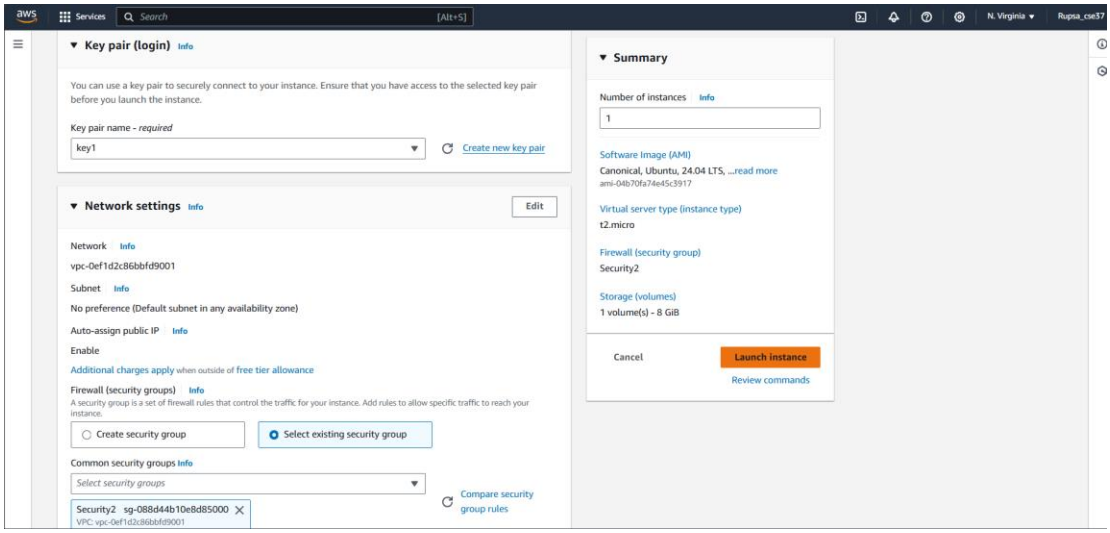


6. Now go back to EC2 dashboard & click on “Launch Instance”.

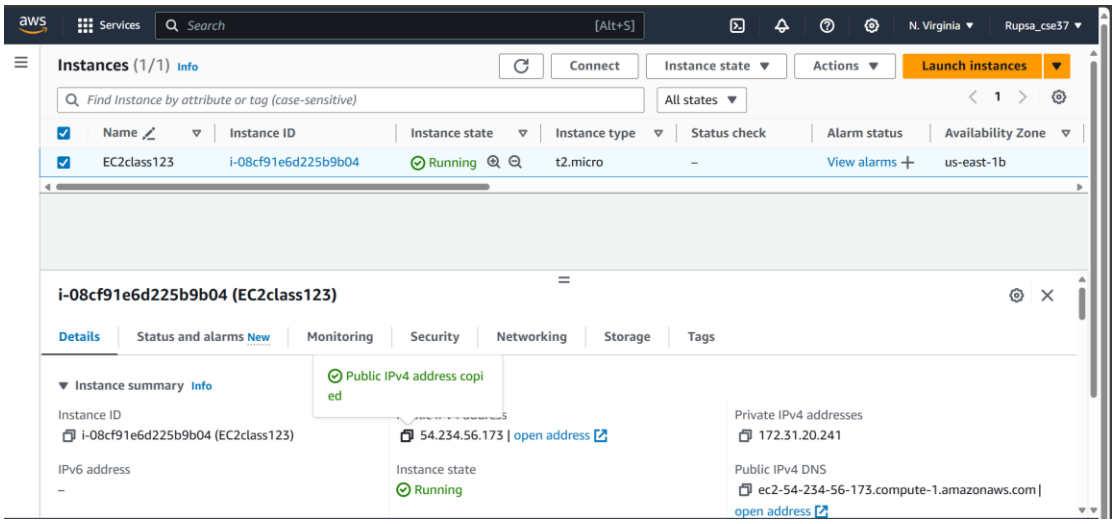


7. Now write the name, choose the platform, keypair and click on “Select existing security group” that is Security2.Then finally click on “Launch Instance”.

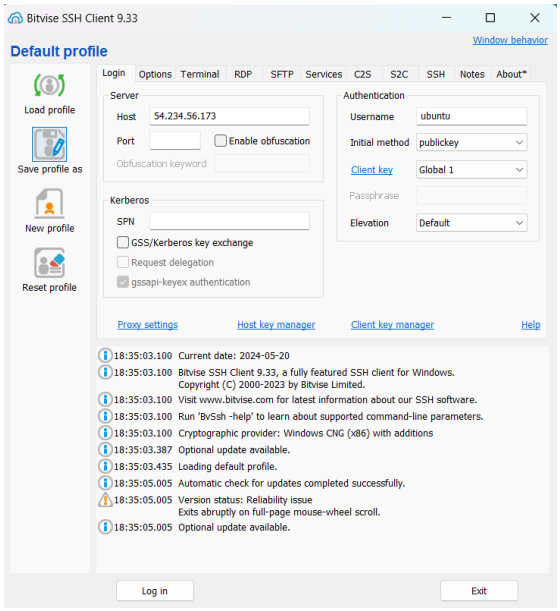


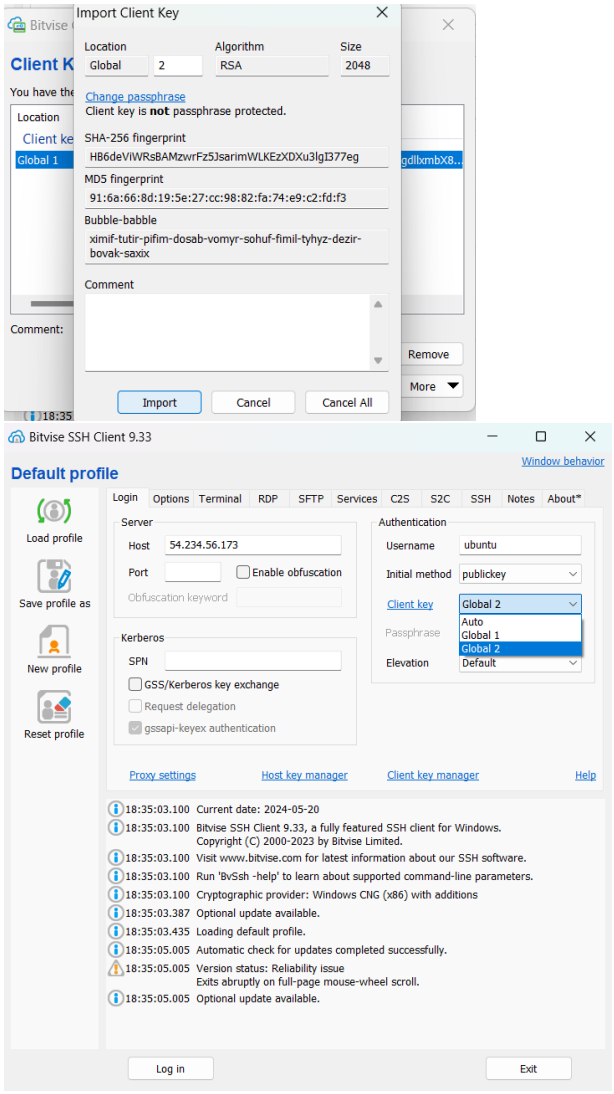


8. The instance is successfully created and select it then copy the Public IPv4 address & paste it in Bitvise.

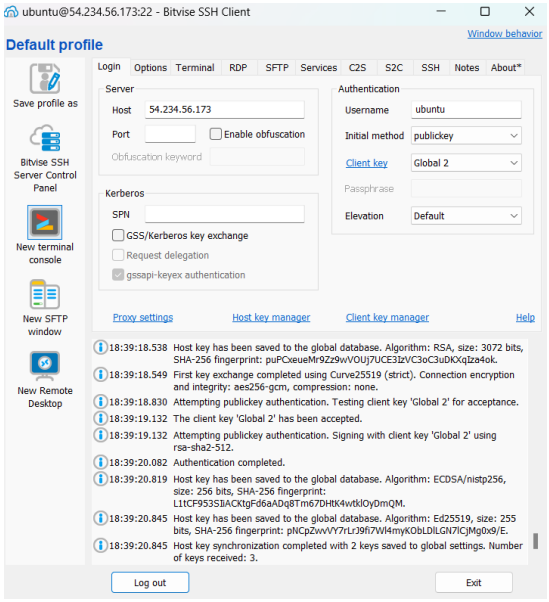


9. Now In Bitvise paste it and import the key from Client key manager then click on “Log in”.





10. After successful log in click on “New terminal console” and write the following commands.



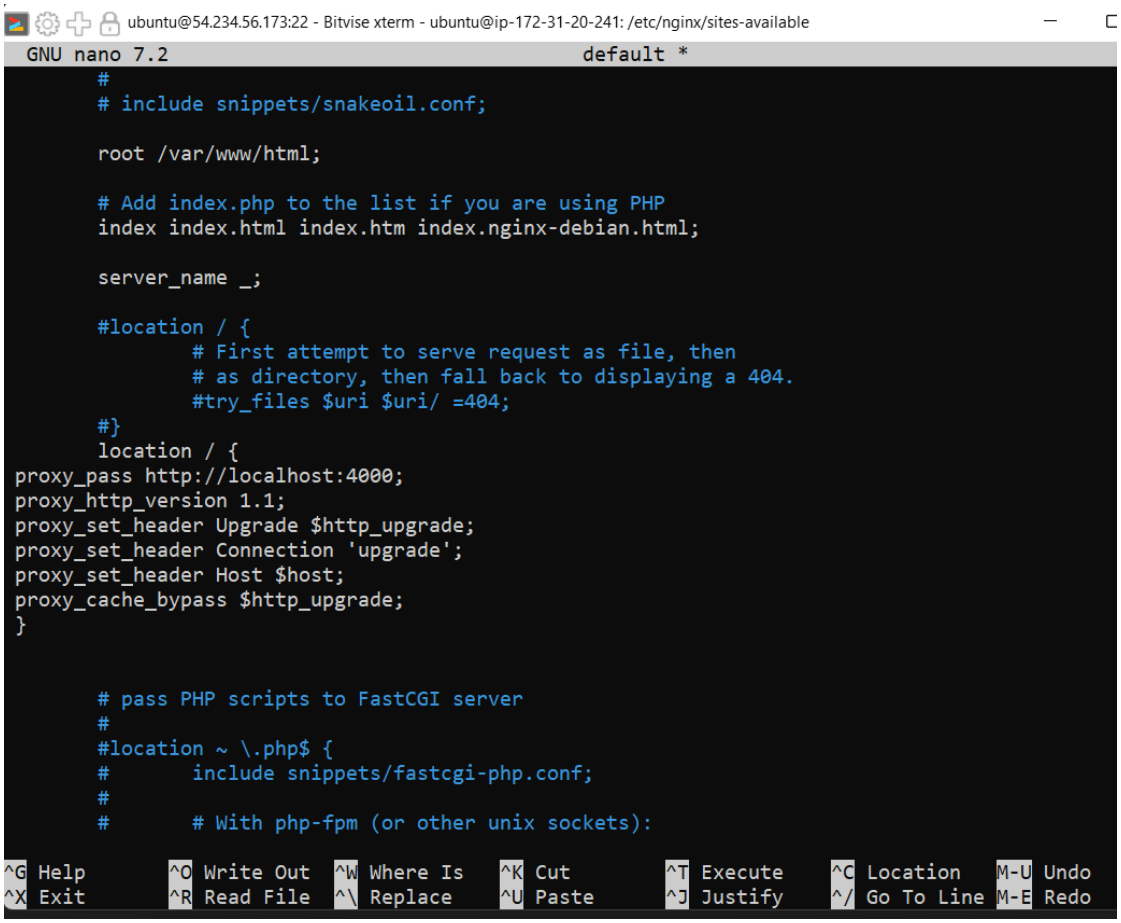
```
ubuntu@ip-172-31-20-241:~$ sudo apt-get update
Get:34 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 c-n-f Metadata [116 B]
Fetched 28.3 MB in 5s (5305 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-20-241:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
User sessions running outdated binaries:
 ubuntu @ session #3: sshd[1012,1123]
 ubuntu @ user manager service: systemd[1017]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-20-241:~$ sudo apt-get install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  nginx-common
Suggested packages:
  fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  nginx nginx-common
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 552 kB of archives.
After this operation, 1596 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
User sessions running outdated binaries:
 ubuntu @ session #3: sshd[1012,1123]
 ubuntu @ user manager service: systemd[1017]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-20-241:~$ nginx -v
nginx version: nginx/1.24.0 (Ubuntu)
ubuntu@ip-172-31-20-241:~$ curl -SL https://deb.nodesource.com/setup_18.x|sudo -E bash -
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  3154  100  3154    0     0  48058      0 --:--:-- --:--:-- --:--:-- 48523
2024-05-20 13:22:42 - Installing pre-requisites
Reading package lists... Done
2024-05-20 13:22:49 - Repository configured successfully. To install Node.js, run: apt-get install nodejs -y
ubuntu@ip-172-31-20-241:~$ sudo apt-get install nodejs
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-20-241:~$ nodejs -v
v18.20.2
ubuntu@ip-172-31-20-241:~$ cd /
ubuntu@ip-172-31-20-241:/$ pwd
/
ubuntu@ip-172-31-20-241:/$ sudo git clone https://github.com/Rupsa1037/Rup_AWS1.git
Cloning into 'Rup_AWS1'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 10 (delta 2), reused 5 (delta 0), pack-reused 0
Receiving objects: 100% (10/10), 49.08 KiB | 7.01 MiB/s, done.
Resolving deltas: 100% (2/2), done.
ubuntu@ip-172-31-20-241:/$ ls
Rup_AWS1  bin.usr-is-merged  dev  home  lib.usr-is-merged  lost+found  mnt  proc  run
bin       boot               etc  lib   lib64             media       opt  root  sbin
ubuntu@ip-172-31-20-241:/$ cd Rup_AWS1/
ubuntu@ip-172-31-20-241:/Rup_AWS1$ ls
'New Text Document.txt'   index.js   package-lock.json   package.json
ubuntu@ip-172-31-20-241:/Rup_AWS1$ sudo npm i
ubuntu@ip-172-31-20-241:/Rup_AWS1$ npm -v
10.5.0
ubuntu@ip-172-31-20-241:/Rup_AWS1$ node index.js
Started server
```

11. Then stop the server and open another “New terminal console’ & write the following commands.

```
Last login: Mon May 20 13:11:25 2024 from 202.78.234.236
ubuntu@ip-172-31-20-241:~$ cd /
ubuntu@ip-172-31-20-241:/$ pwd
/
ubuntu@ip-172-31-20-241:/$ cd etc/nginx/sites-available/
ubuntu@ip-172-31-20-241:/etc/nginx/sites-available$ sudo nano default
```




```
ubuntu@ip-172-31-20-241:/etc/nginx/sites-available$ sudo systemctl restart nginx
ubuntu@ip-172-31-20-241:/etc/nginx/sites-available$
```

Then on the another terminal start the server by “node index.js”.

```
ubuntu@ip-172-31-20-241:/Rup_AWS1$ node index.js
Started server
^C
ubuntu@ip-172-31-20-241:/Rup_AWS1$ node index.js
Started server
```

12. Paste the IPv4 in the browser to host the file.