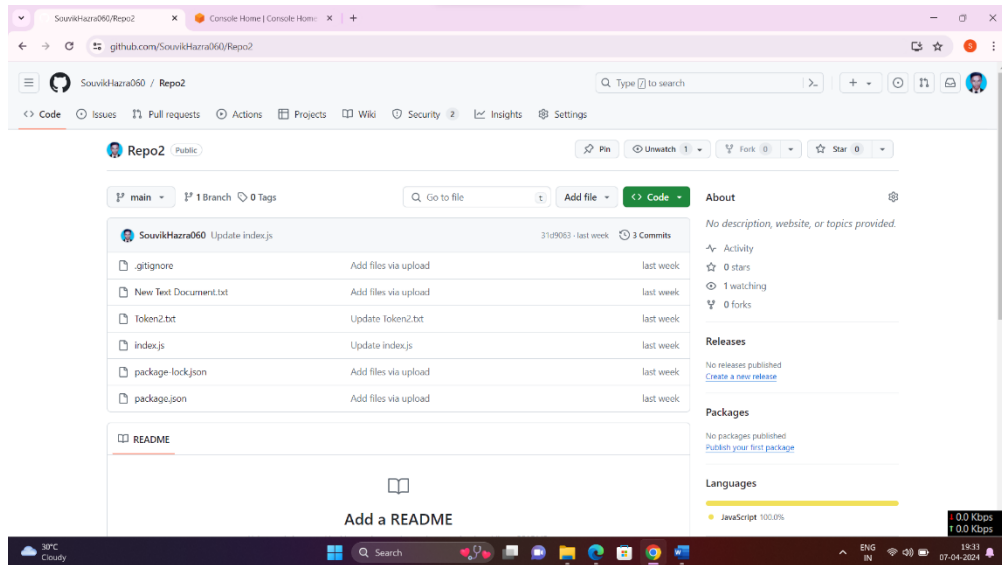


Assignment 9

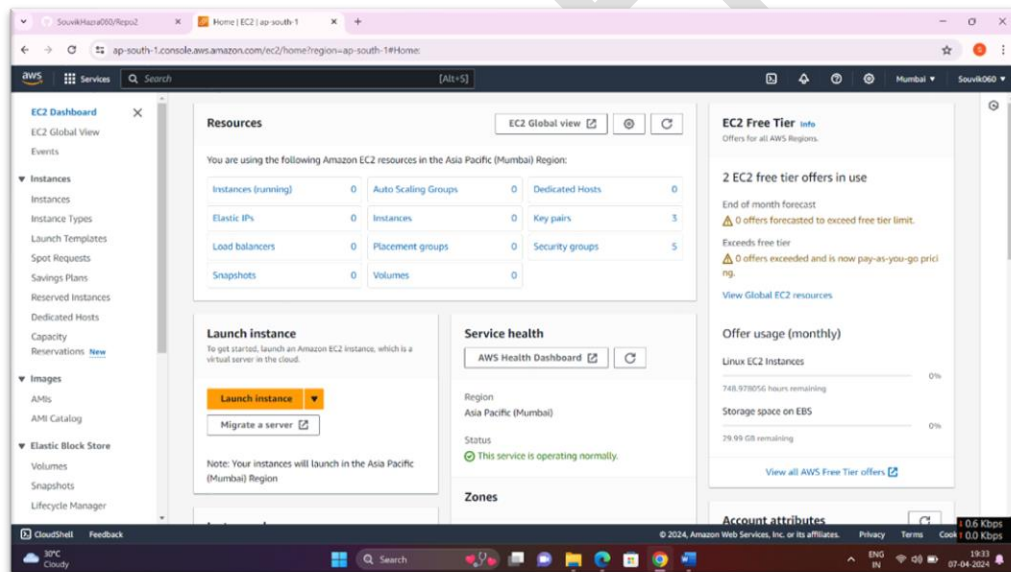
Problem Statement : Deploy a project from GitHub on EC2.

Steps:

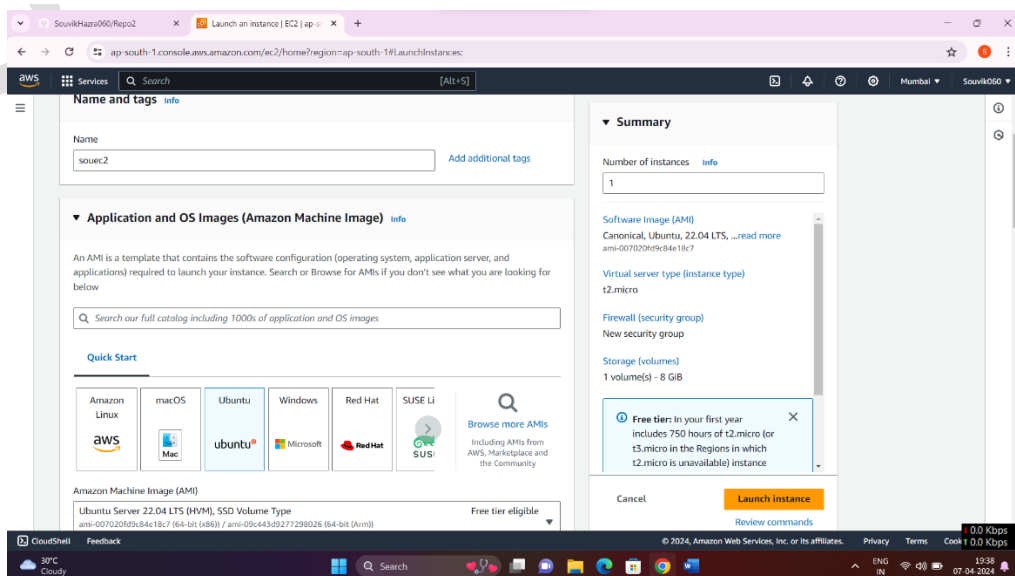
1) At first upload all files within a repository to Github.



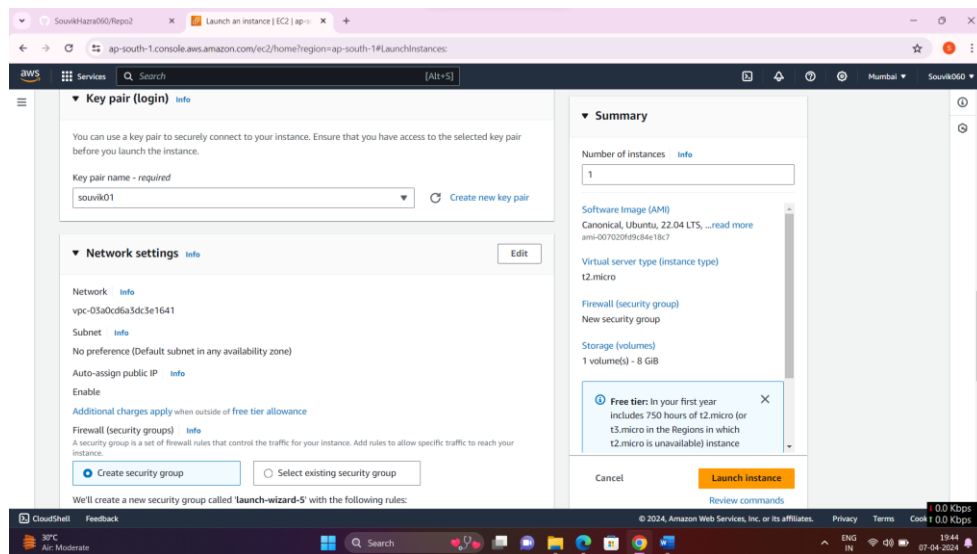
2) Now open EC2 in aws console and goto Launch Instance.



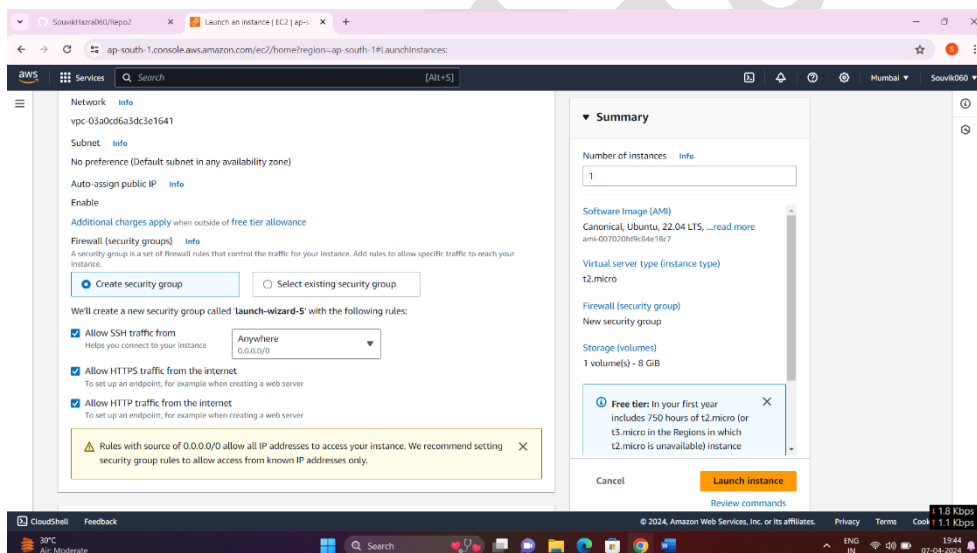
3) Now give name of server which should be unique and select Ubuntu application and OS image inside quick start.



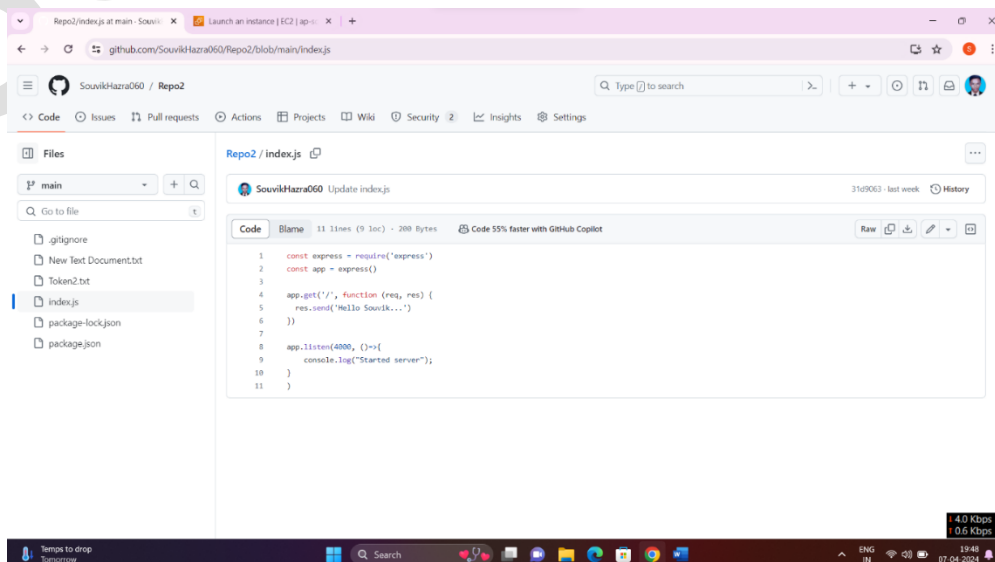
4) Now generate new key pair(give name (unique) and now click Create Key Pair and also download the document) or use the existing key pair . In this case we use existing one.



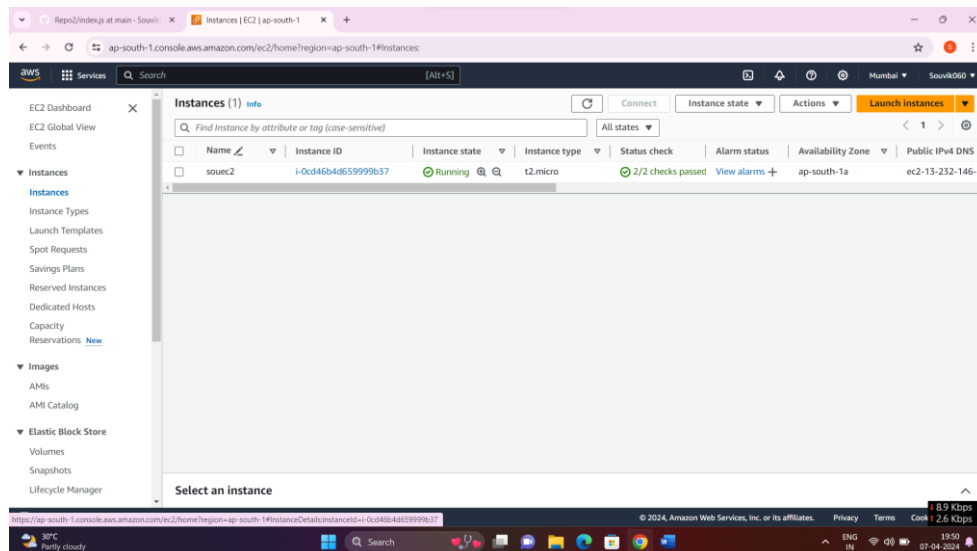
5) After that select Create security group and click all checkboxes bellow it. After it click on Launch Instance.



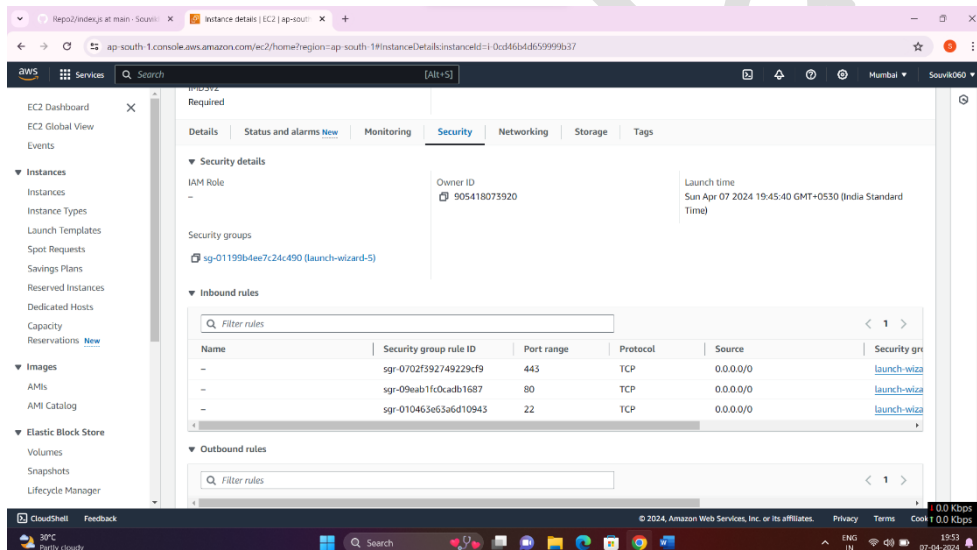
6) Now go to Github that project and then to index.js ,there we can see 4000 in app.listen which is port no that will be added to custom TCP.



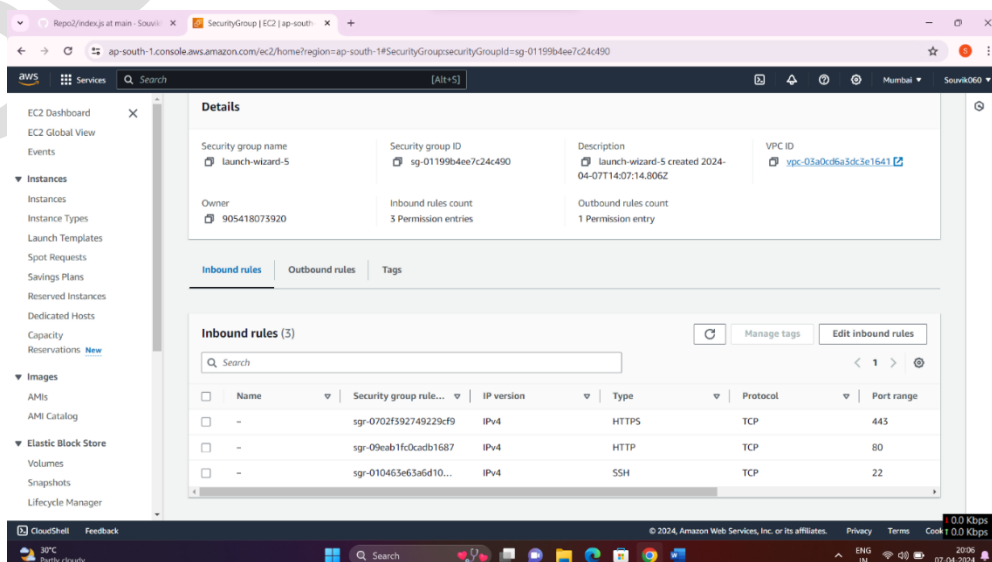
7) After it go back to EC2 instances then select that instance newly made and click on Security groups.



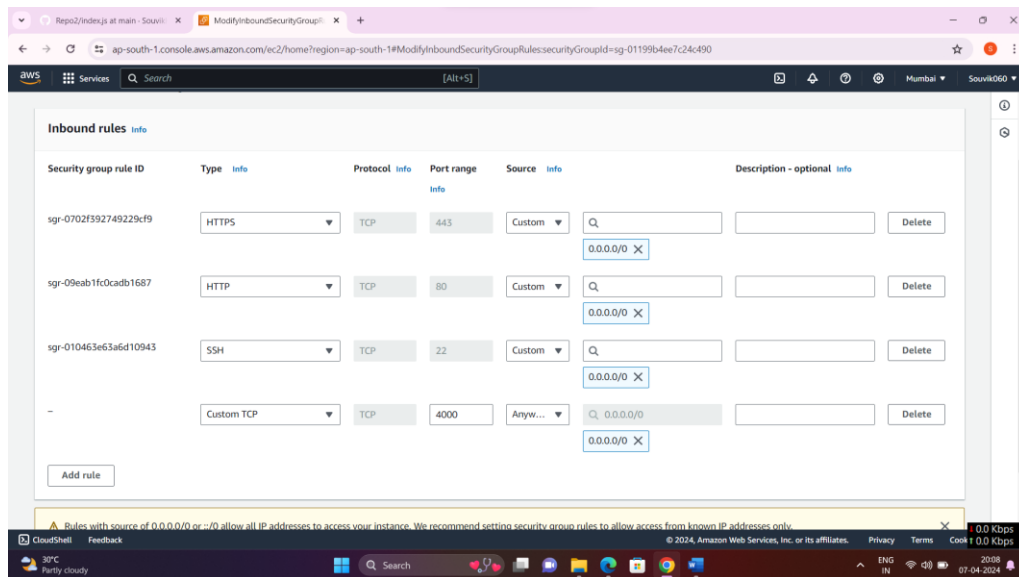
8) In security group we can see that Inbound rules and also Outbound rules



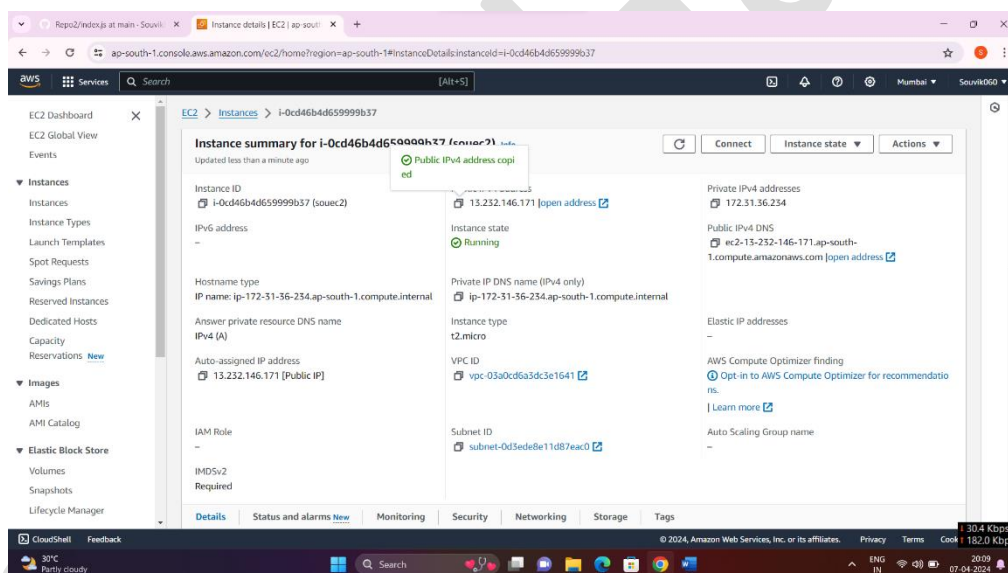
9) Then click on the link below "Security groups" and then in Inbound rules section click on Edit inbound rules.



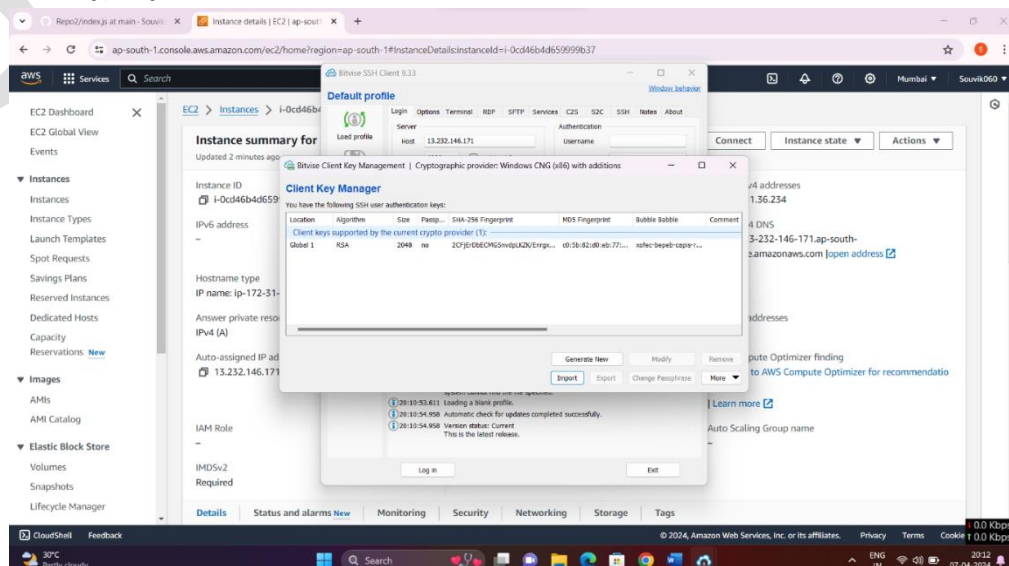
10) Then in inbound rules, click on “Add rule” select Custom TCP ,give port range 4000 and select 0.0.0.0/0. After it click on Save rules right side bottom corner.



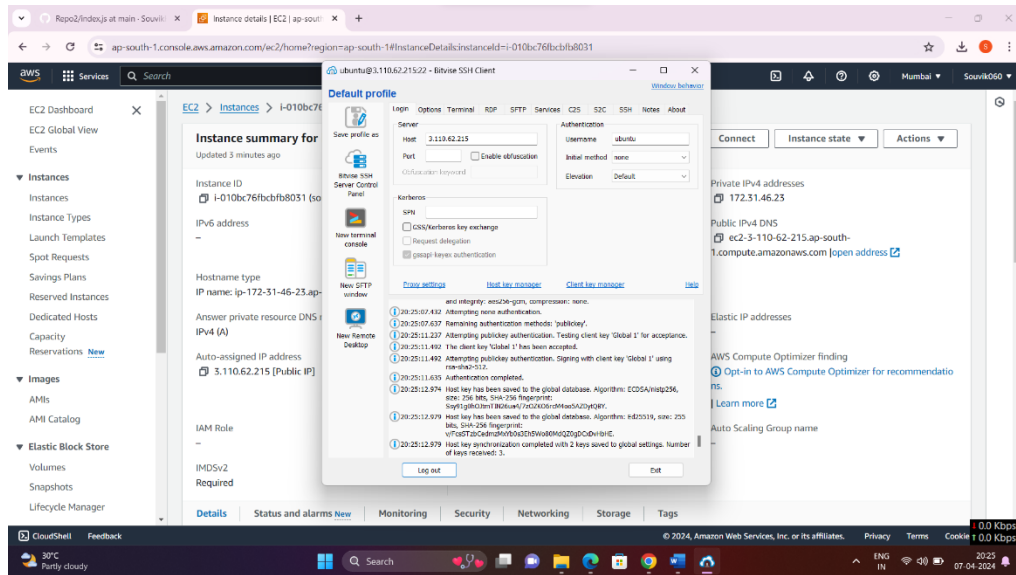
11) Now go back to EC2 recently created instance and copy the public IPv4 address.



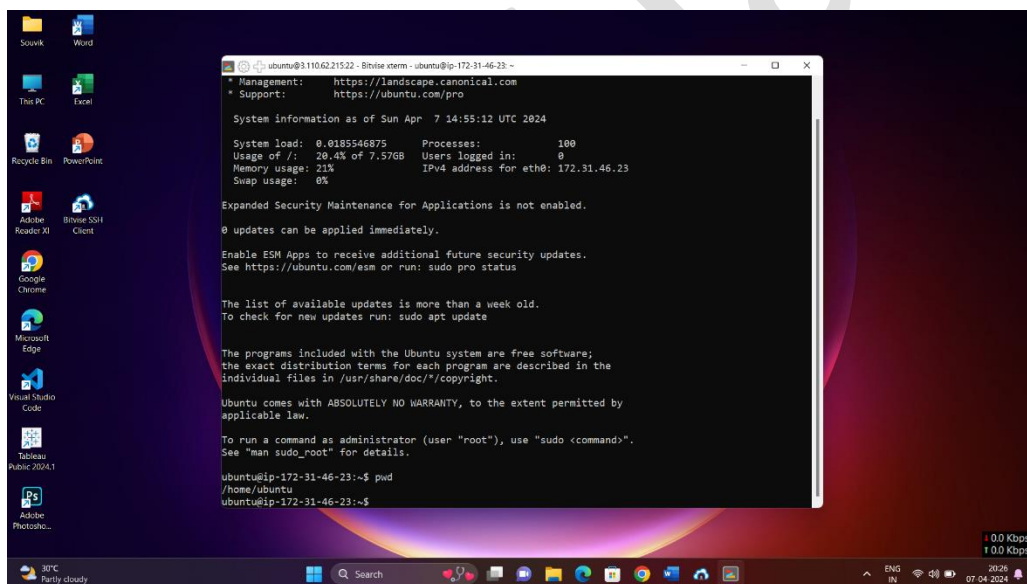
12) Now open Bitwise SSH Client, go to Client key manager and import that downloaded or saved (previously) key.



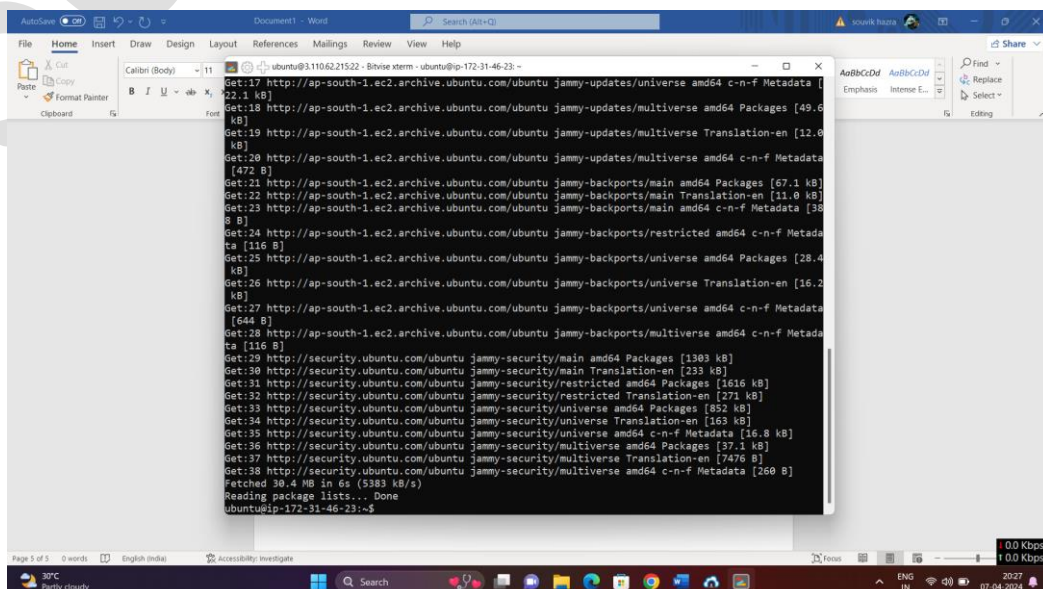
13) After that paste copied IPv4 public address in host and then login and also do Accept and Save, also set publickey,global1 then click on ok.



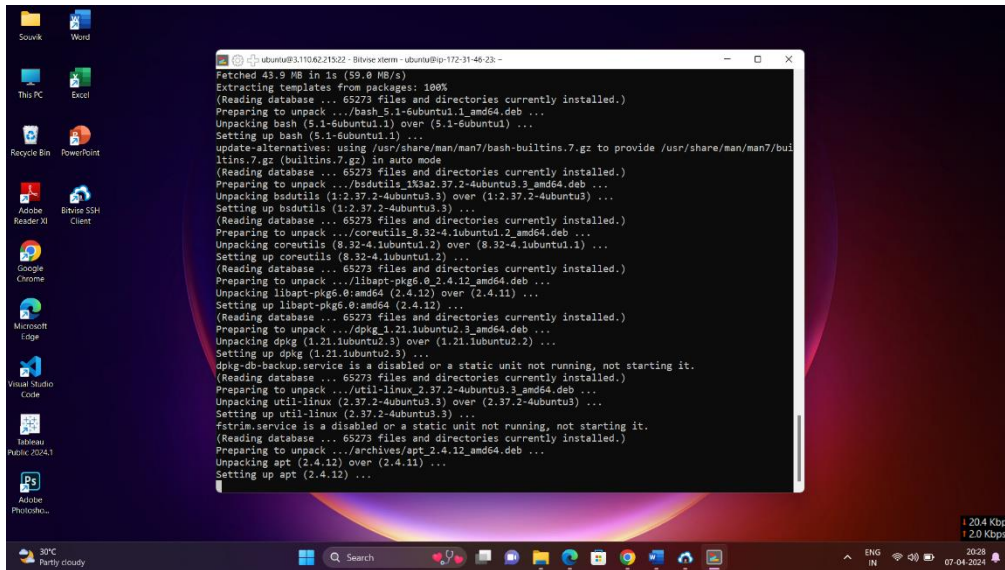
14) Now open a new terminal and write pwd to check working directory, we are in ubuntu.



15) Now write command sudo apt-get update to fetch all packages.

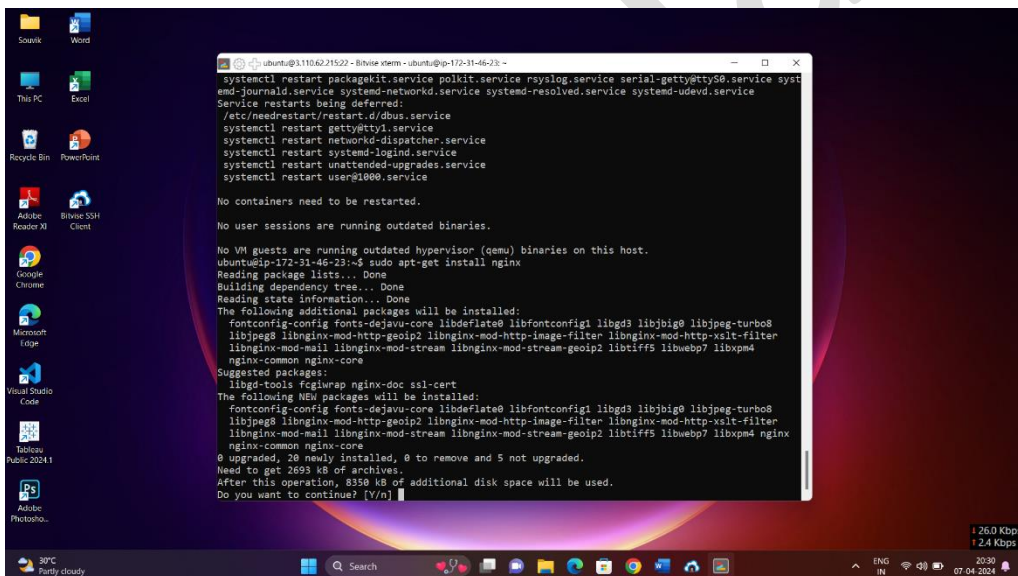


16) Write sudo apt-get upgrade to upgrade all outdated packages.



```
ubuntu@31106221522: ~$ sudo apt-get upgrade
Fetched 43.9 MB in 1s (59.0 MB/s)
Extracting templates from packages: 100%
(Reading database ... 65273 files and directories currently installed.)
Preparing to unpack .../bash.5.1-6ubuntu1.1_amd64.deb ...
Unpacking bash (5.1-6ubuntu1.1) over (5.1-6ubuntu1) ...
Setting up bash (5.1-6ubuntu1.1) ...
update-alternatives: using /usr/share/man/man7/bash-builtin.7.gz to provide /usr/share/man/man7/bu
ldash.7.gz (builtin.7.gz) in auto mode
(Reading database ... 65273 files and directories currently installed.)
Preparing to unpack .../bsdutils_1:3.2.3-2ubuntu3.3_amd64.deb ...
Unpacking bsdutils (1:3.2.3-2ubuntu3.3) over (1:2.37.2-4ubuntu3) ...
Setting up bsdutils (1:2.37.2-4ubuntu3.3) ...
(Reading database ... 65273 files and directories currently installed.)
Preparing to unpack .../coreutils_8.32-4.1ubuntu1.2_amd64.deb ...
Unpacking coreutils (8.32-4.1ubuntu1.2) over (8.32-4.1ubuntu1.1) ...
Setting up coreutils (8.32-4.1ubuntu1.2) ...
(Reading database ... 65273 files and directories currently installed.)
Preparing to unpack .../libapt-pkg6.0_2.4.12_amd64.deb ...
Unpacking libapt-pkg6.0:amd64 (2.4.12) over (2.4.11) ...
Setting up libapt-pkg6.0:amd64 (2.4.12) ...
(Reading database ... 65273 files and directories currently installed.)
Preparing to unpack .../dpkg_1.21.1ubuntu2.3_amd64.deb ...
Unpacking dpkg (1.21.1ubuntu2.3) over (1.21.1ubuntu2.2) ...
Setting up dpkg (1.21.1ubuntu2.3) ...
dpkg-db-backup.service is a disabled or a static unit not running, not starting it.
(Reading database ... 65273 files and directories currently installed.)
Preparing to unpack .../util-linux_2.37.2-4ubuntu3.3_amd64.deb ...
Unpacking util-linux (2.37.2-4ubuntu3.3) over (2.37.2-4ubuntu3) ...
Setting up util-linux (2.37.2-4ubuntu3.3) ...
fatrm.service is a disabled or a static unit not running, not starting it.
(Reading database ... 65273 files and directories currently installed.)
Preparing to unpack .../archives/apt_2.4.12_amd64.deb ...
Unpacking apt (2.4.12) over (2.4.11) ...
Setting up apt (2.4.12) ...
```

17) Now write command sudo apt-get install nginx to install webserver.



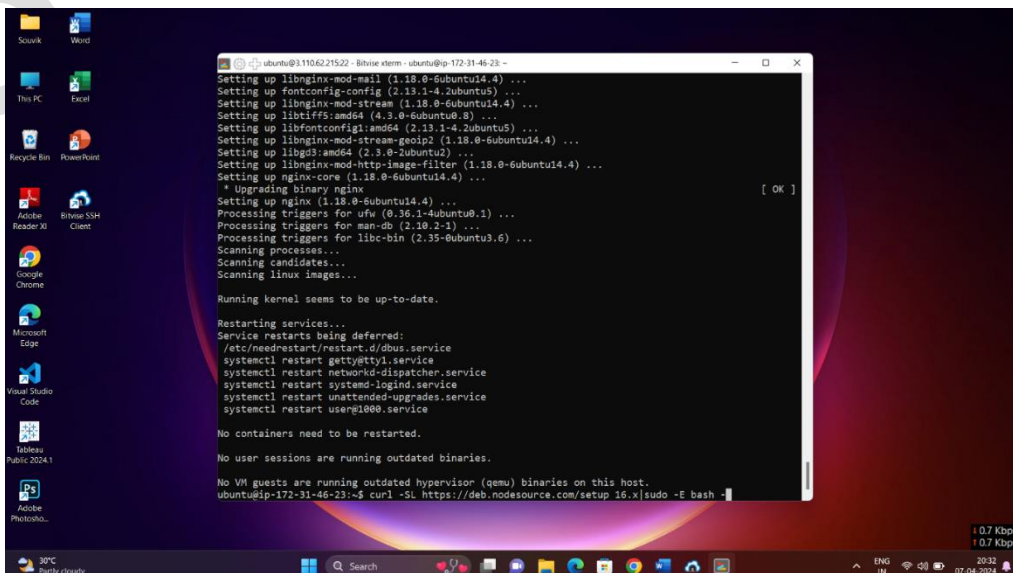
```
ubuntu@31106221522: ~$ sudo apt-get install nginx
systemctl restart packagekit.service polkit.service rsyslog.service serial-getty@ttyS0.service syst
emd-journald.service systemd-networkd.service systemd-resolved.service systemd-udev.service syst
Service restarts being deferred:
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service
systemctl restart user@1000.service

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-46-23:~$ sudo apt-get install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libfontconfig1 libgd3 libjpeg8 libjpeg-turbo8
  libpng8 libpng-mod-http-geoip2 libpng-mod-http-image-filter libpng-mod-http-xslt-filter
  libpng-mod-mail libpng-mod-stream libpng-mod-stream-geoip2 libtiff5 libwebp7 libxpm4
  nginx-common nginx-core
Suggested packages:
  libgd-tools fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  fontconfig-config fonts-dejavu-core libdeflate0 libfontconfig1 libgd3 libjpeg8 libjpeg-turbo8
  libpng8 libpng-mod-http-geoip2 libpng-mod-http-image-filter libpng-mod-http-xslt-filter
  libpng-mod-mail libpng-mod-stream libpng-mod-stream-geoip2 libtiff5 libwebp7 libxpm4 nginx
  nginx-common nginx-core
0 upgraded, 20 newly installed, 0 to remove and 5 not upgraded.
Need to get 2693 kB of archives.
After this operation, 8358 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

18) Now to execute javascript, we need to install nodejs. For this just write the command curl -SL https://deb.nodesource.com/setup_16.x|sudo -E bash -



```
ubuntu@31106221522: ~$ curl -SL https://deb.nodesource.com/setup_16.x|sudo -E bash -
Setting up libpng-mod-mail (1.18.0-6ubuntu14.4) ...
Setting up fontconfig-config (2.13.1-4.2ubuntu5) ...
Setting up libpng-mod-stream (1.18.0-6ubuntu14.4) ...
Setting up libtiff5:amd64 (4.3.0-6ubuntu8) ...
Setting up libfontconfig1:amd64 (2.13.1-4.2ubuntu5) ...
Setting up libpng-mod-stream-geoip2 (1.18.0-6ubuntu14.4) ...
Setting up libgd3:amd64 (2.3.0-2ubuntu2) ...
Setting up libpng-mod-http-image-filter (1.18.0-6ubuntu14.4) ...
Setting up nginx-core (1.18.0-6ubuntu14.4) ...
* Upgrading binary nginx
Setting up nginx (1.18.0-6ubuntu14.4) ...
Processing triggers for ufw (0.36-1-4ubuntu0.1) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.6) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Running kernel seems to be up-to-date.

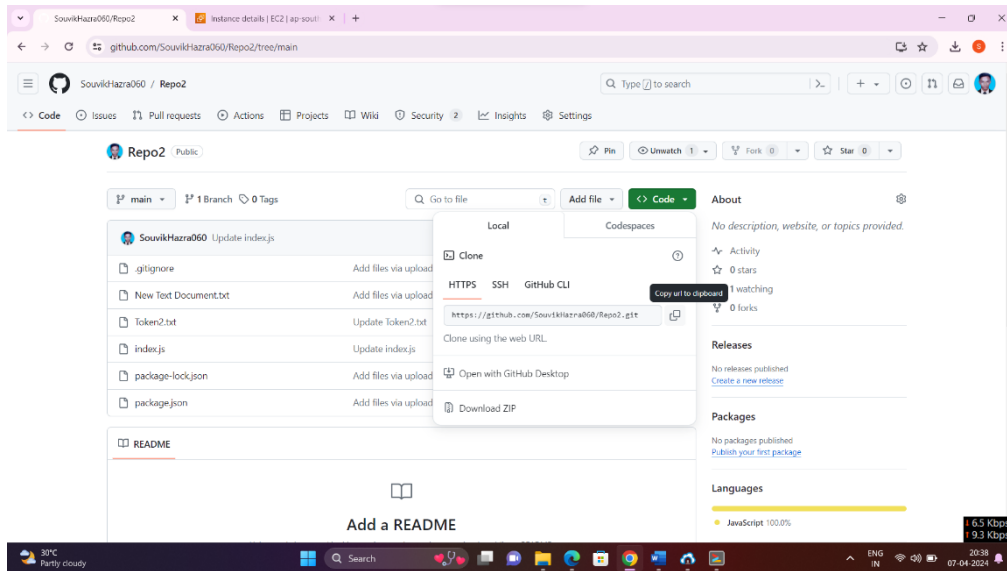
Restarting services...
Service restarts being deferred:
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service
systemctl restart user@1000.service

No containers need to be restarted.

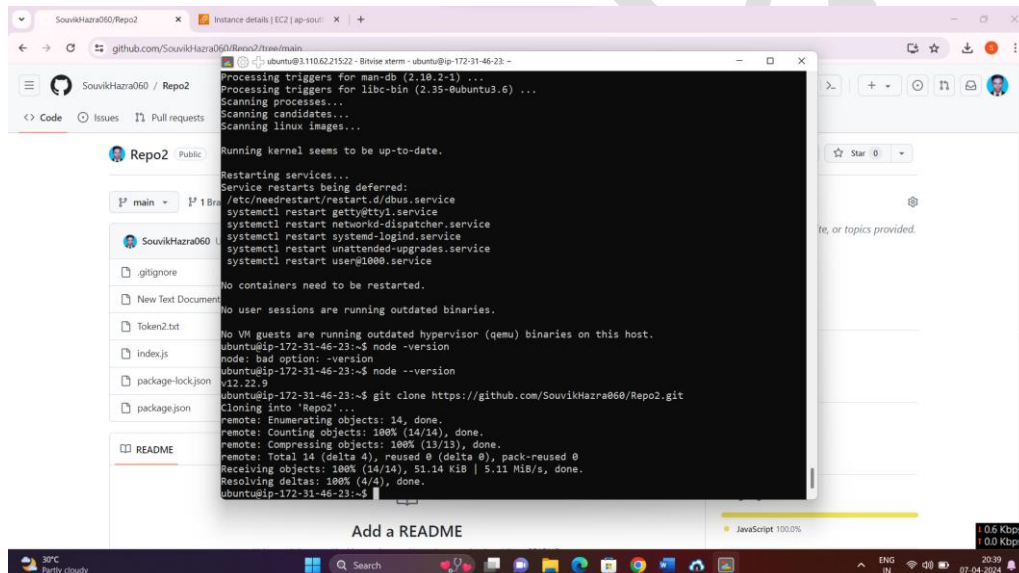
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-46-23:~$ curl -SL https://deb.nodesource.com/setup_16.x|sudo -E bash -
```

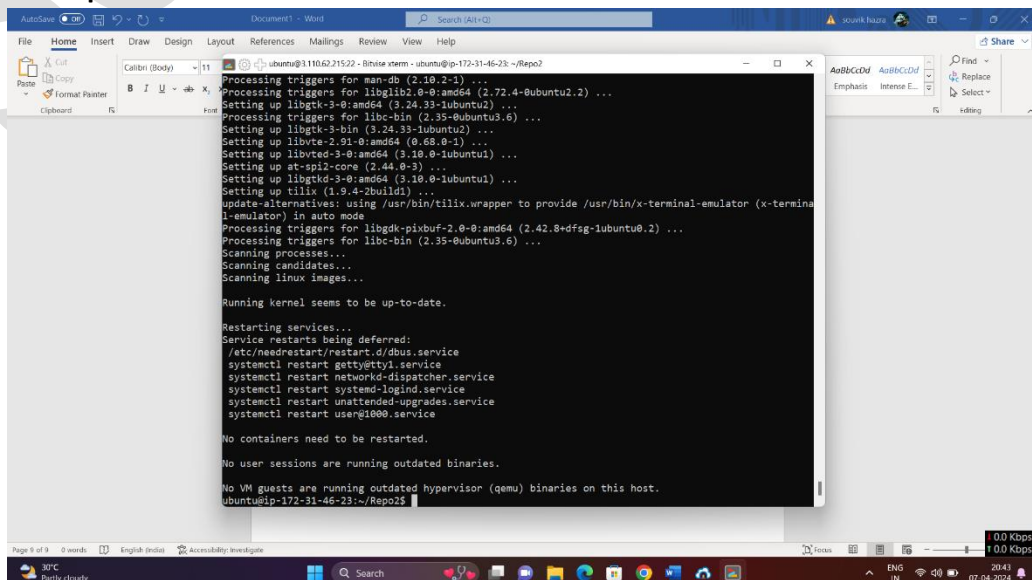

21) Now go back to github enter into repository and in Code section copy Https url.



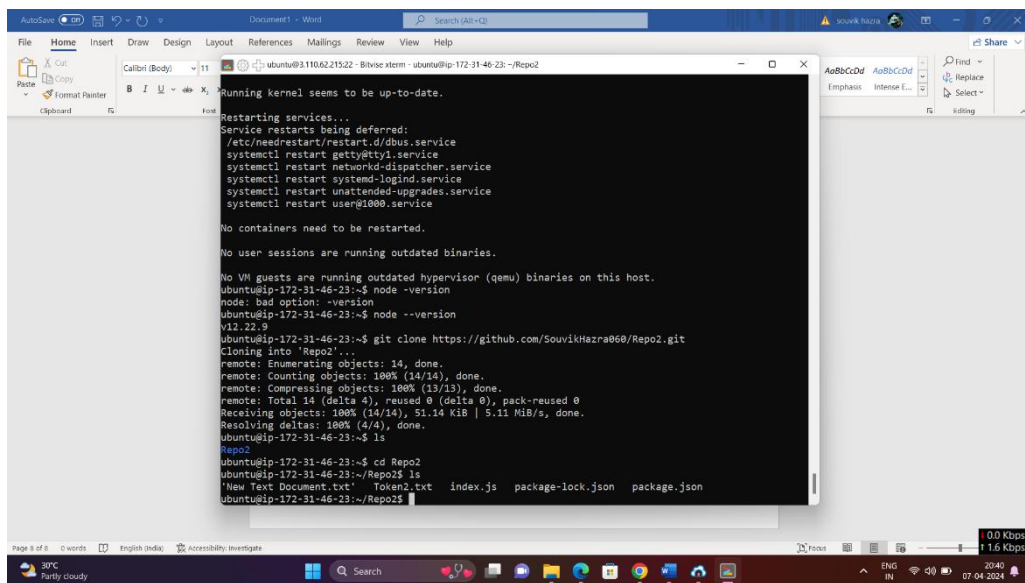
22) Now write git clone and paste that url and write 'ls' command to see if project has been cloned or not.



23) After it write command cd (project name) to enter into project and then ls to see what files have been uploaded.



24) Now to execute node command we have to install node packet manager(npm). So write npm install.



```
Running kernel seems to be up-to-date.
Restarting services...
Service restarts being deferred:
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service
systemctl restart user@1000.service
No containers need to be restarted.
No user sessions are running outdated binaries.
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-46-23:~$ node -version
node: bad option: -version
ubuntu@ip-172-31-46-23:~$ node --version
v12.22.9
ubuntu@ip-172-31-46-23:~$ git clone https://github.com/SouvikHazra860/Repo2.git
Cloning into 'Repo2'...
remote: Enumerating objects: 14, done.
remote: Counting objects: 100% (14/14), done.
remote: Compressing objects: 100% (13/13), done.
remote: Total 14 (delta 4), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (14/14), 51.14 KiB | 5.11 MiB/s, done.
Resolving deltas: 100% (4/4), done.
ubuntu@ip-172-31-46-23:~$ ls
Repo2
ubuntu@ip-172-31-46-23:~$ cd Repo2
ubuntu@ip-172-31-46-23:~/Repo2$ ls
'New Text Document.txt'  Token2.txt  index.js  package-lock.json  package.json
ubuntu@ip-172-31-46-23:~/Repo2$
```

25) Now write node index.js .Server will started.

```
ubuntu@ip-172-31-46-23:~/Repo2$ node index.js
Started server
```

26) After it again copy public IPv4 address in EC2 instance and then paste it in another tab(Incognito) url section. We get “Welcome to nginx!”



27) At last in url at end write :4000 to get our website .



28) Now to close server in new terminal do (ctrl+c) to stop server and at last logout.

- In this way we have deployed a project from GitHub to EC2.