

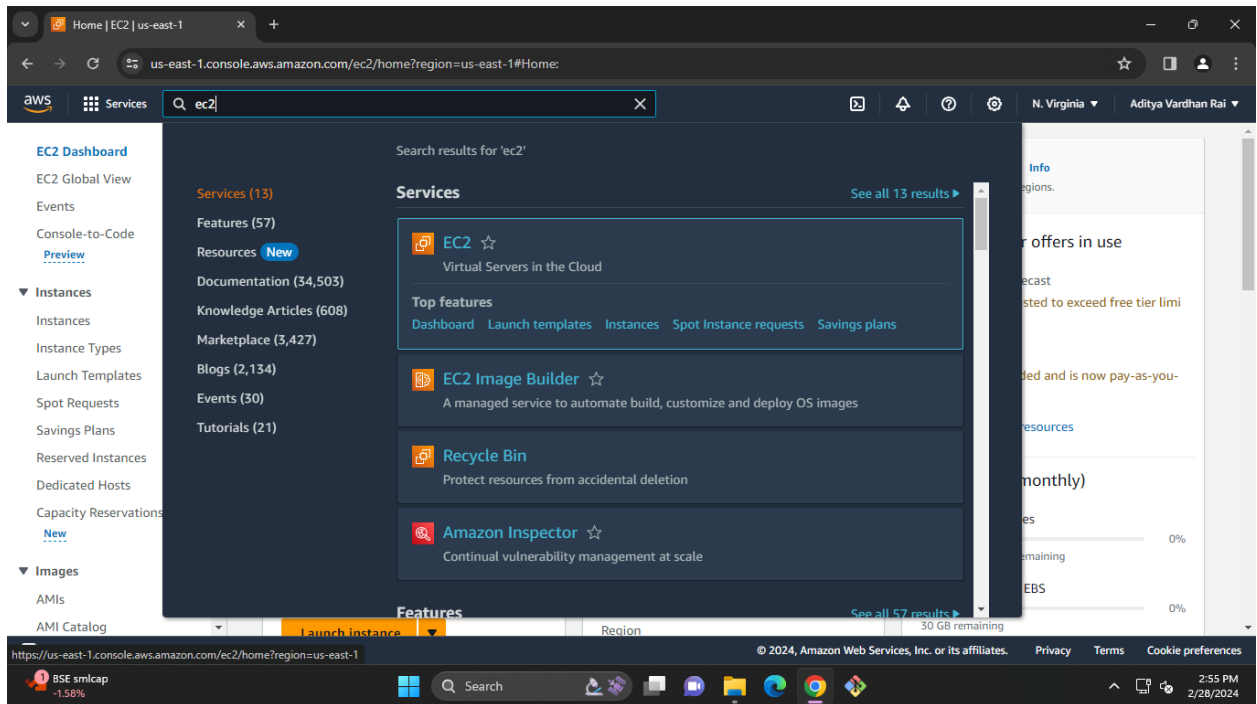
Assignment-7

PROBLEM STATEMENT :

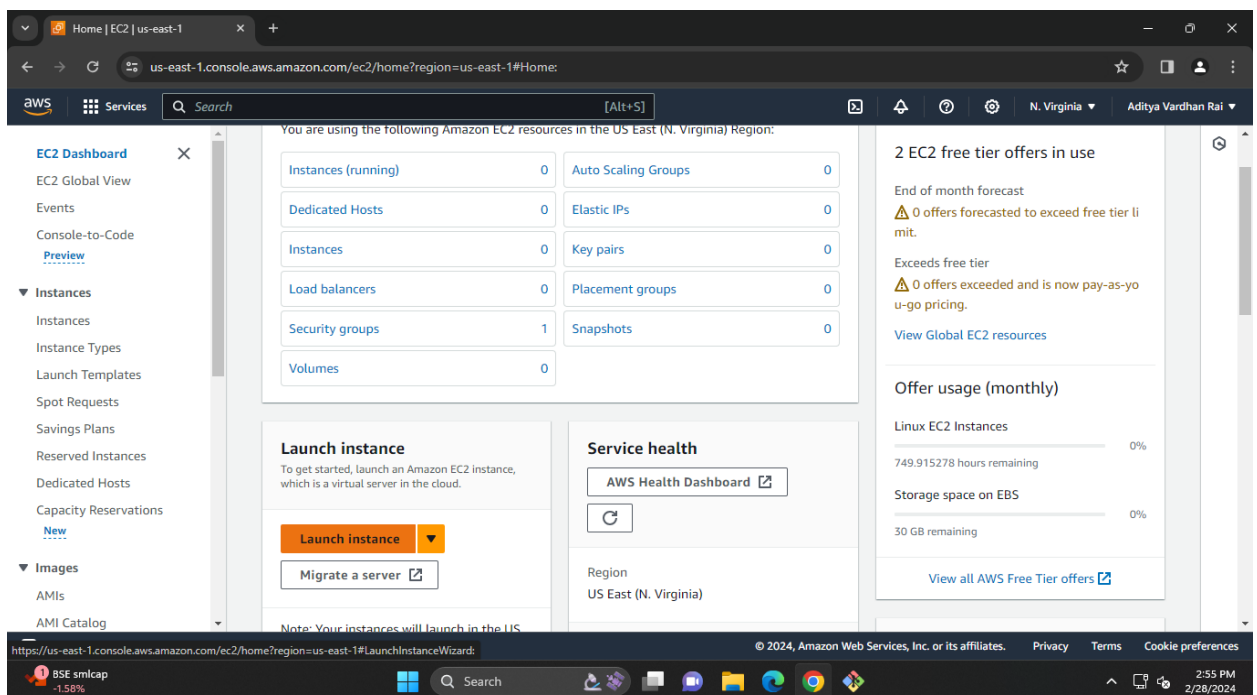
Hosting a website on EC2.

Steps to host a website on EC2->

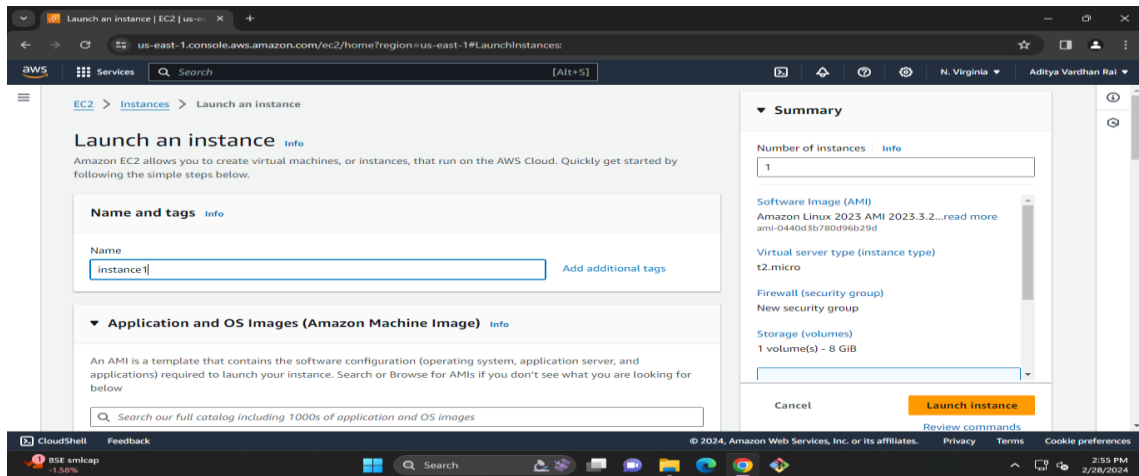
1. Sign up for an AWS account, search for 'EC2' then click on it.



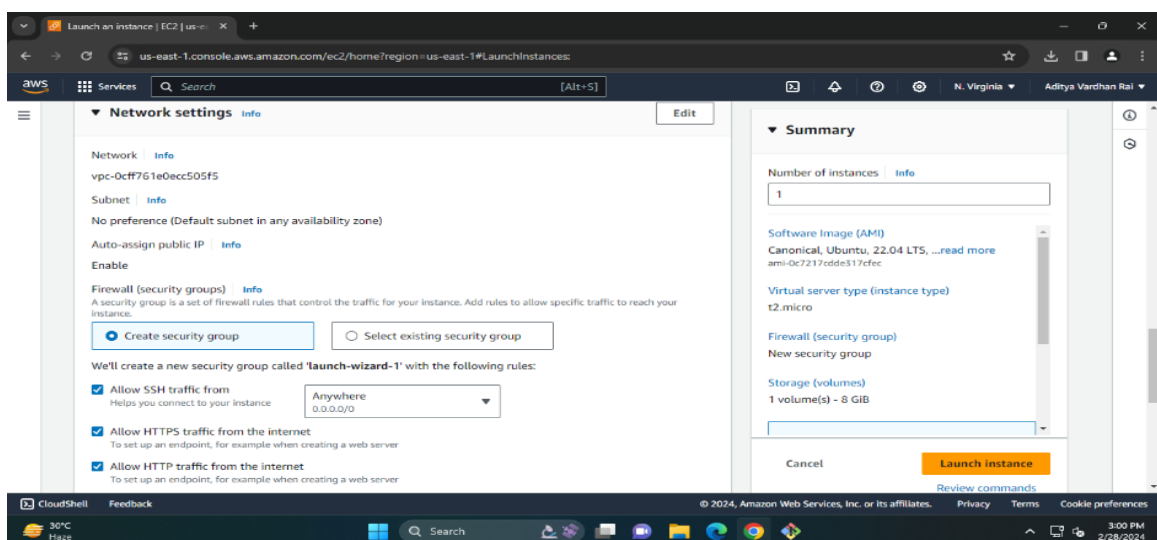
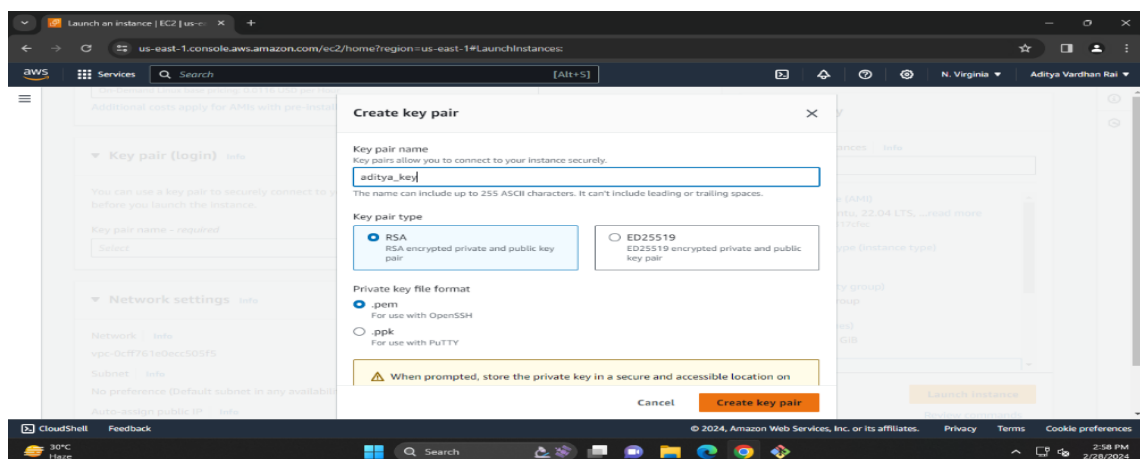
2. Click on 'Launch instance'.



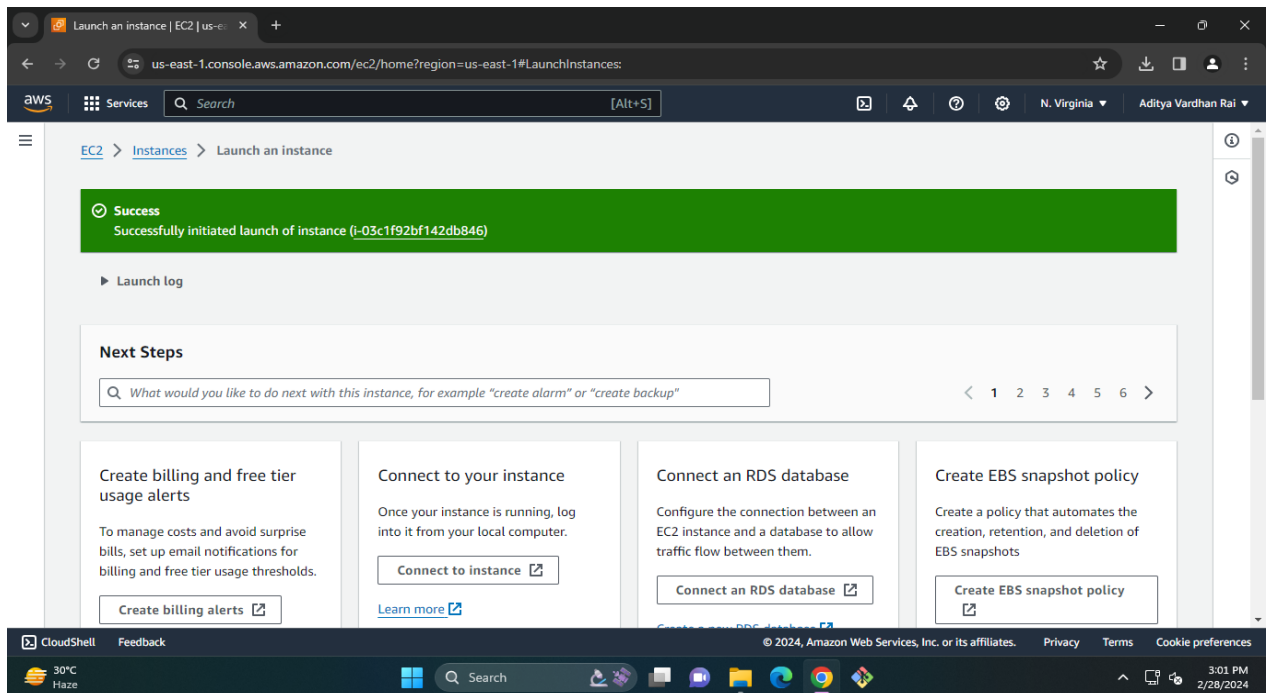
3. Fill up the required details->'Name' and under 'Application and OS Images' select 'ubuntu'.



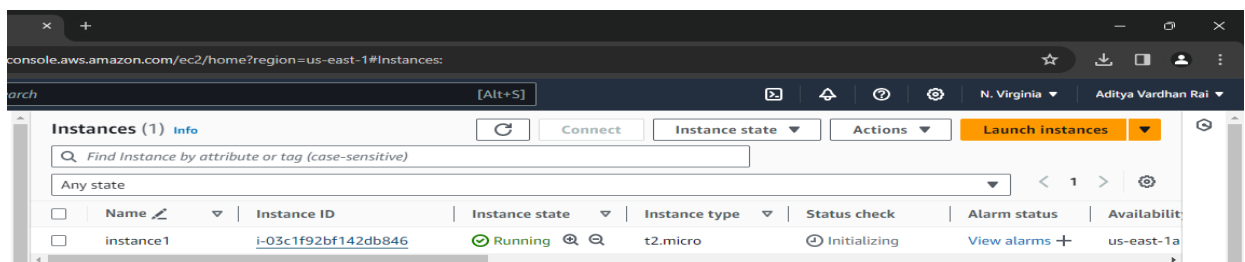
In key pair(login)', click on 'Create new key pair', give 'key pair name' and create it. Under 'Network Settings' tick on all checkboxes. Launch instance.



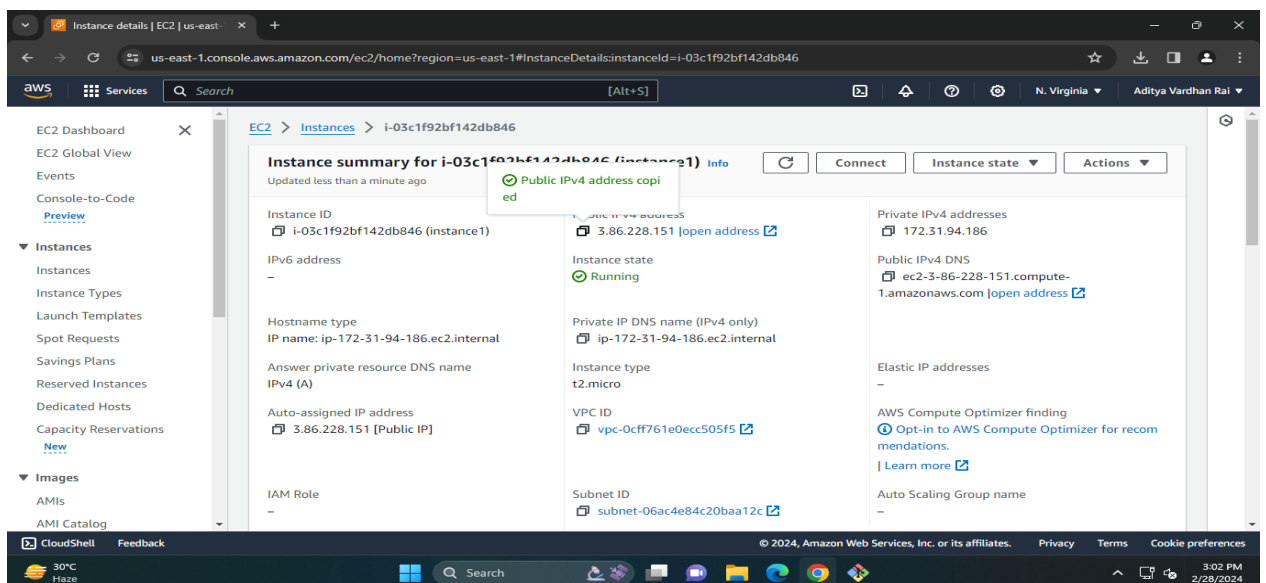
4. 'instance1' instance is created successfully. Then click on 'Instances'.



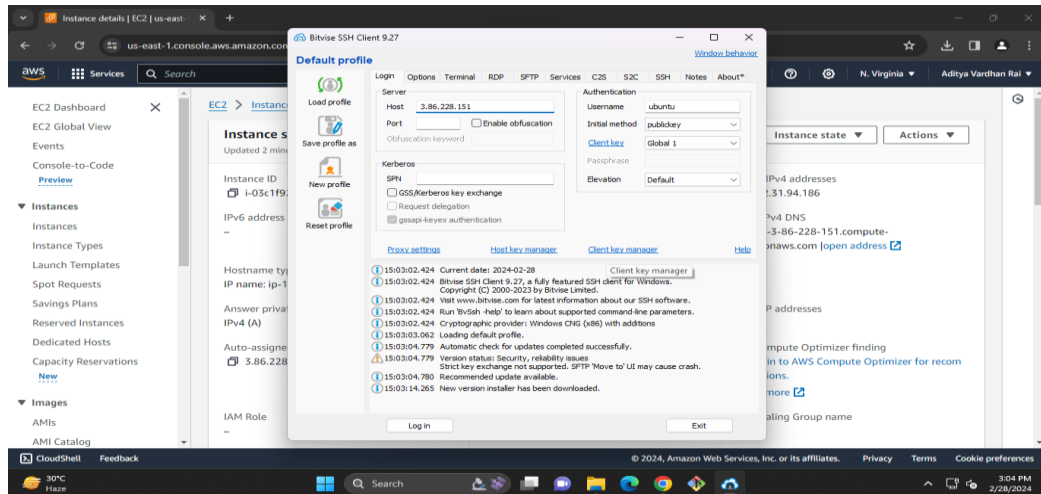
5. Now click on 'Instance ID'.



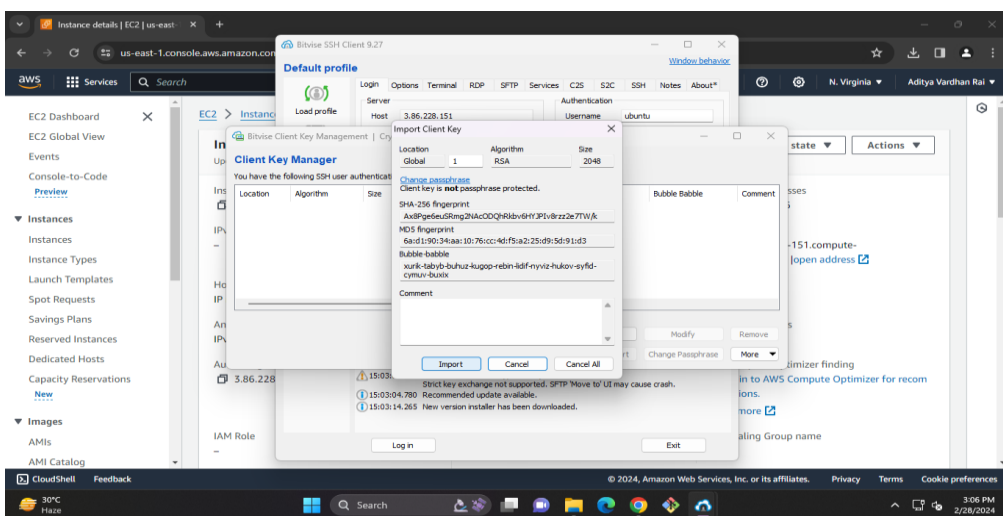
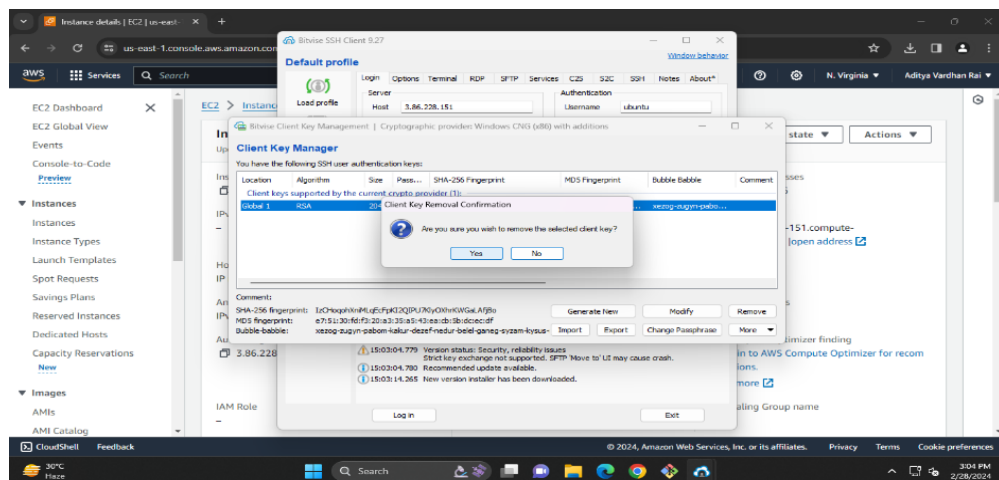
6. Click on 'Public IPV4 address' & copy it then open 'Bitwise SSH Client'.



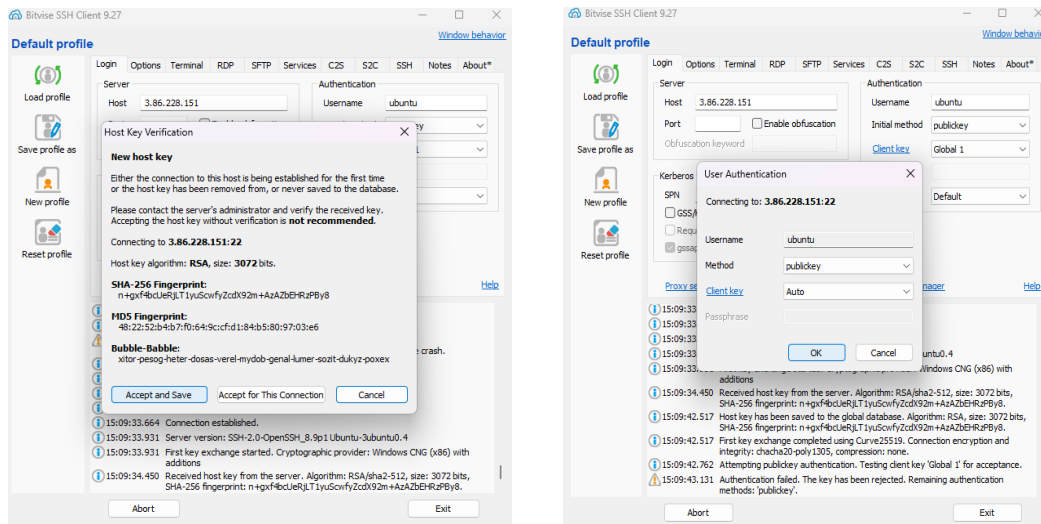
7. In Bitvise SSH Client, paste the 'Public IPV4 address' 7 click on 'Client key Manager'.



8. Under 'Client key manager', if there is any existing key then remove it and click on 'Import', select the created key 'instance1.pem' then again click on 'Import' and then the key is successfully imported.



9. In 'Bitvise SSH Client', under 'Default profile' give the 'Username' as 'ubuntu' then select Global1 from 'Client Key Manager' then click on 'Log in' & 'Accept and save'.



10. In 'Bitvise SSH client', under 'Default profile' click on 'New terminal console' and there run 'pwd' command to check where we are.

```
ubuntu@3.88.206.65:22 - Bitvise xterm - ubuntu@ip-172-31-19-43: ~
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-19-43:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-19-43:~$
```

In terminal console, type the following commands.-> 'sudo apt-get update', 'sudo apt-get upgrade', 'sudo apt-get install nginx'.

```
ubuntu@ip-172-31-19-43:~$ sudo apt-get update
Reading package lists... Done

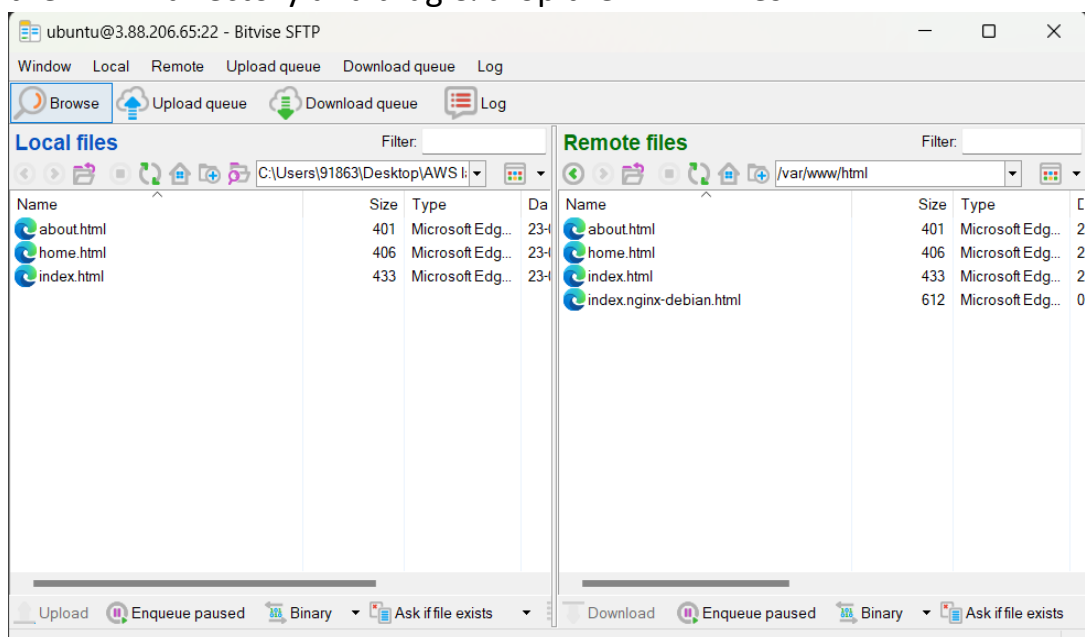
ubuntu@ip-172-31-19-43:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages have been kept back:
  cloud-init linux-aws linux-headers-aws linux-image-aws
The following packages will be upgraded:
  base-files bind9-dnsutils bind9-host bind9-libs binutils binutils-common binutils-x86-64-linux-gnu coreutils iptables less libbinutils libctf-nobfd0 libctf0 libip4tc2
  libip6tc2 libssl3 libuv1 libxml2 libxtables12 login motd-news-config open-vm-tools openssl passwd python-apt-common python3-apt python3-distupgrade tcpdump tzdata
  ubuntu-release-upgrader-core
30 upgraded, 0 newly installed, 0 to remove and 4 not upgraded.
Need to get 13.9 MB of archives.
After this operation, 23.6 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y

ubuntu@ip-172-31-19-43:~$ sudo apt-get install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

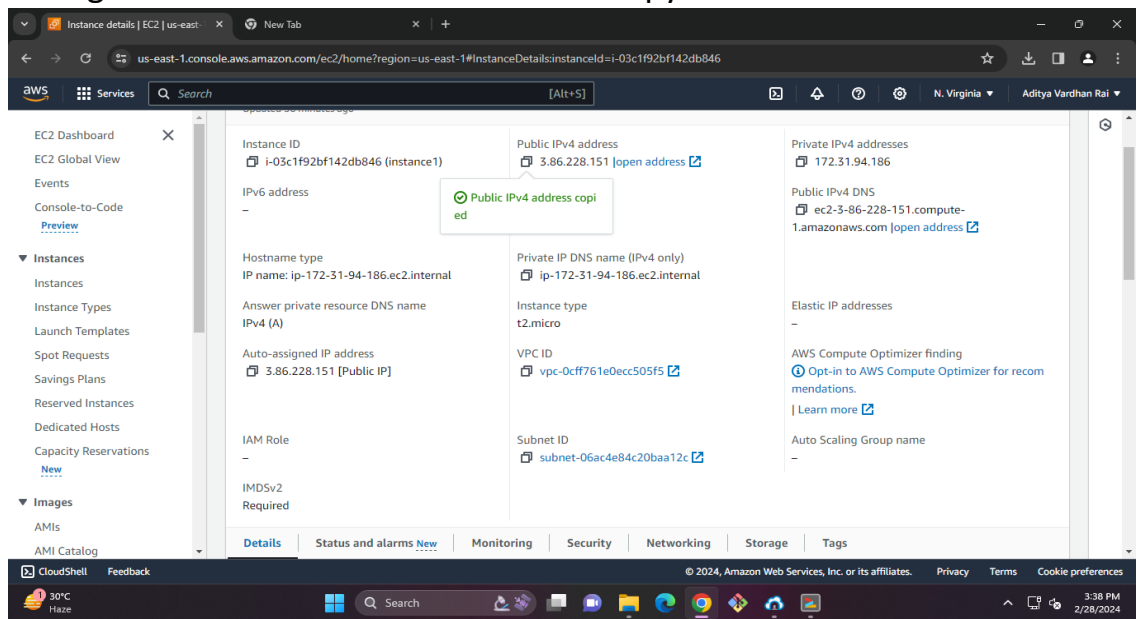
11. Check if www folder appears or not by using 'ls' command inside 'var' folder. Now to give the file access permissions to it go back to the 'www' directory, and type the command 'sudo chmod 777 html' and press 'Enter'.

```
ubuntu@ip-172-31-19-43:/$ cd var
ubuntu@ip-172-31-19-43:/var$ ls
backups cache crash lib local lock log mail opt run snap spool tmp www
ubuntu@ip-172-31-19-43:/var$ cd www
ubuntu@ip-172-31-19-43:/var/www$ ls
html
ubuntu@ip-172-31-19-43:/var/www$ cd html
ubuntu@ip-172-31-19-43:/var/www/html$ ls
index.nginx-debian.html
ubuntu@ip-172-31-19-43:/var/www/html$ cd ..
ubuntu@ip-172-31-19-43:/var/www$ sudo chmod 777 html
ubuntu@ip-172-31-19-43:/var/www$
```

12. Now going back to the 'SFTP Window' under the 'Remote Files' open the HTML directory and drag & drop the HTML files.



13. Now go back to the 'AWS Window' and copy the 'IPv4 address'.



14. In a new window paste the 'IPv4 address'.

