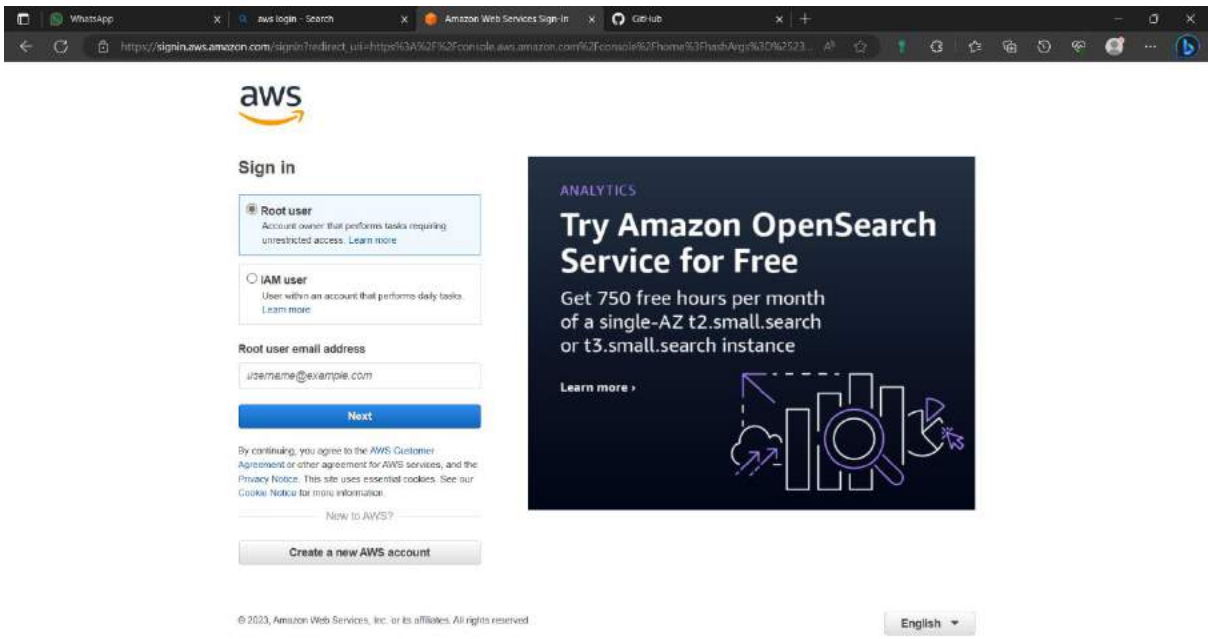
MINI PROJECT ON AWS(AMAZON WEB SERVICES)

IAM Hands-On:

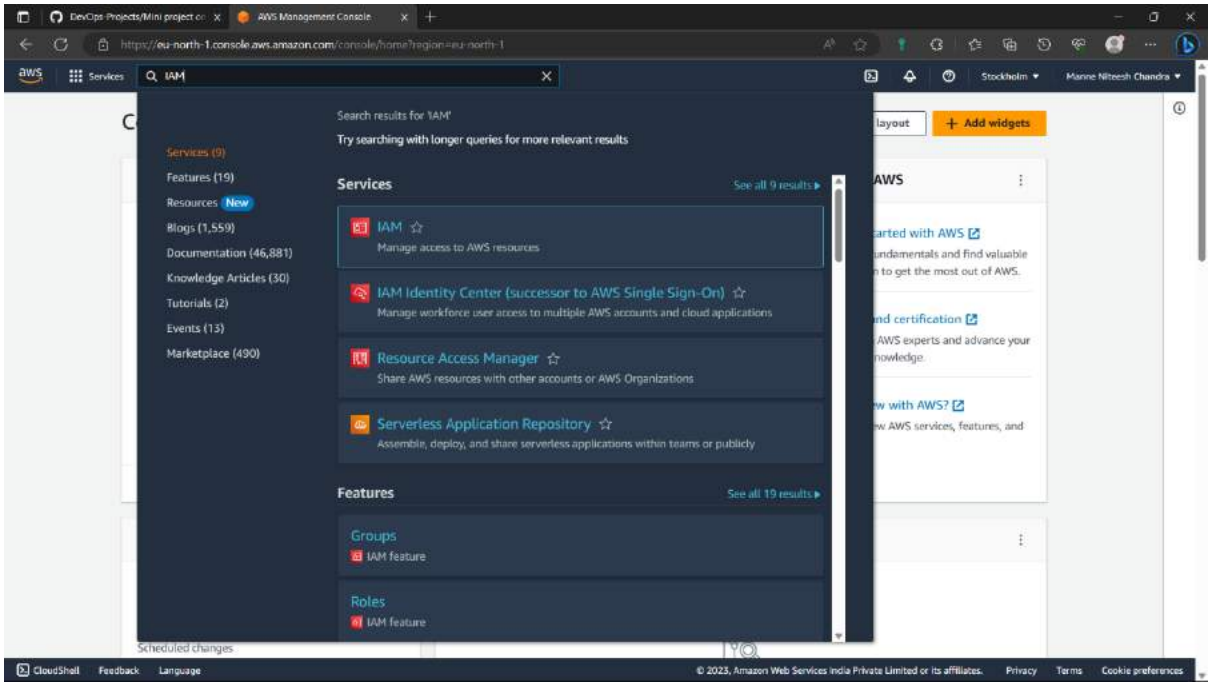
Github link:https://github.com/NiteeshManne?tab=repositories

1)Login to the AWS console as a root user.

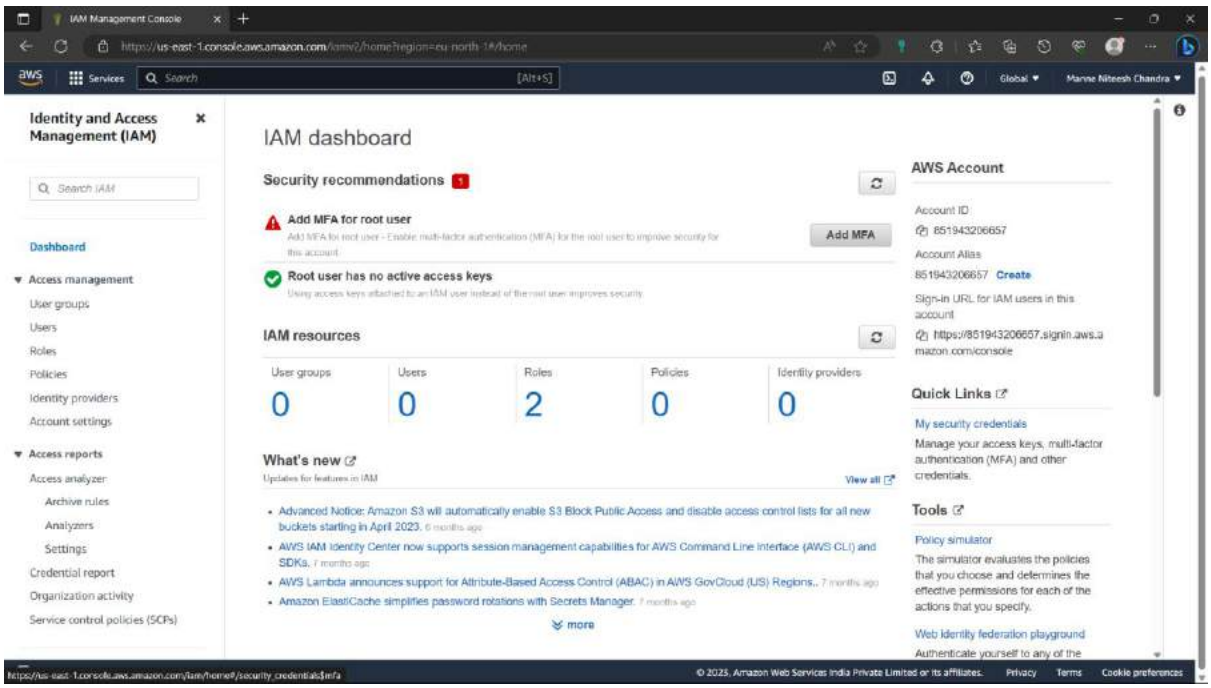


---

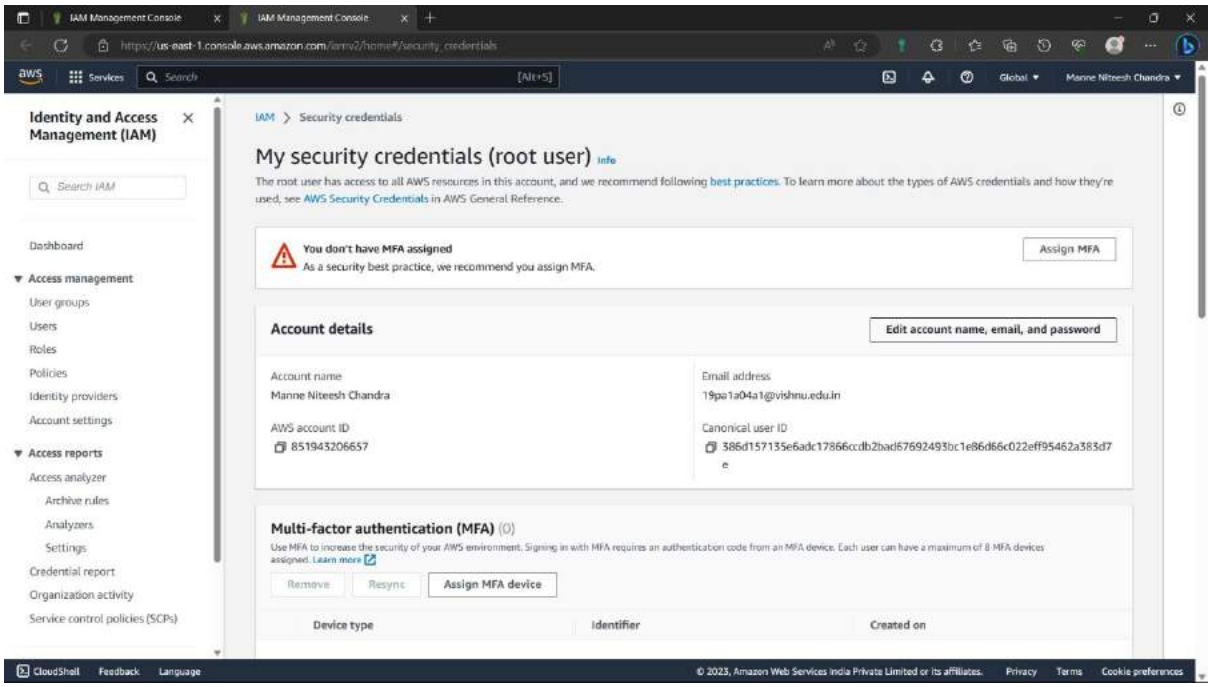
2)Type IAM in search bar and select the IAM option.



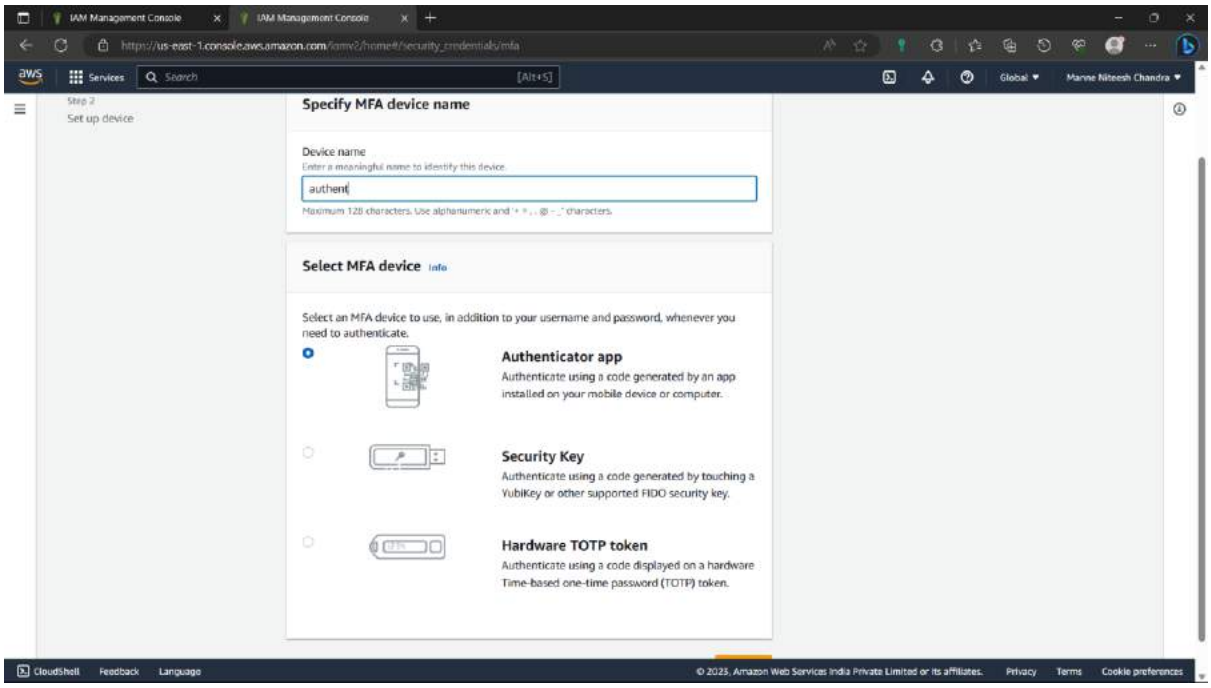
3)Click on add MFA



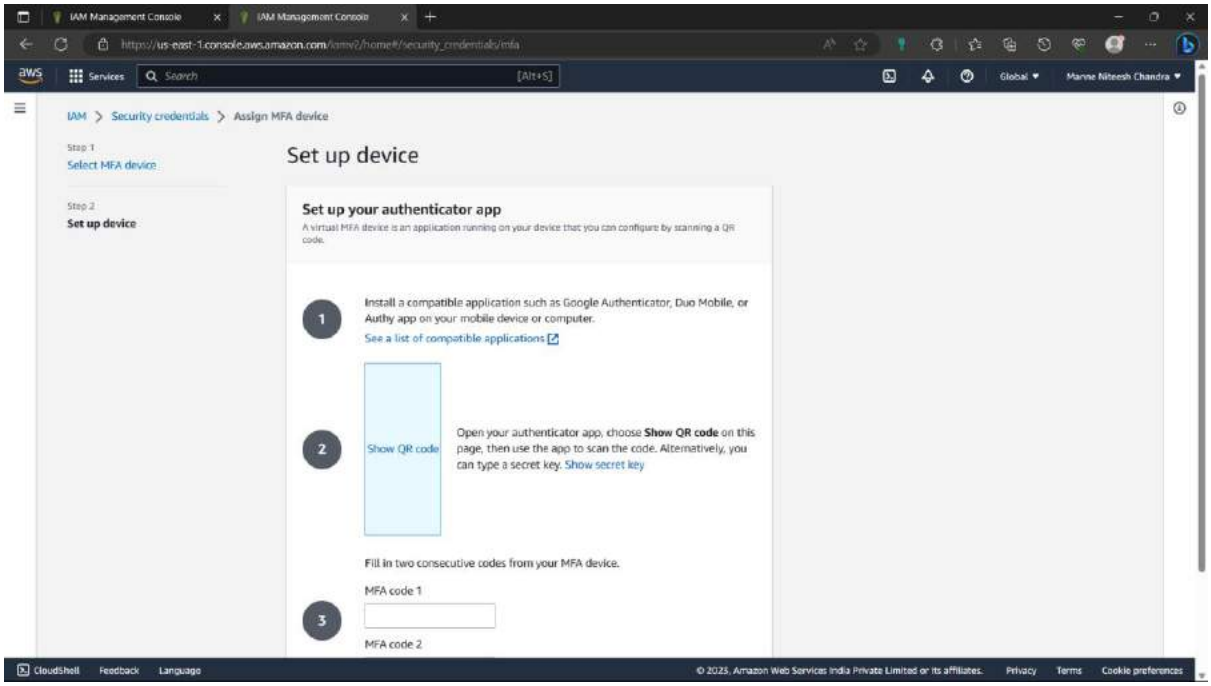
4)It will open My security credentials in a new tab, Click on “Assign MFA” option.



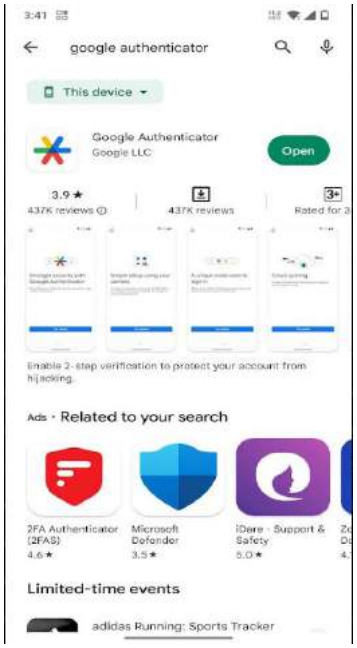
5)Now specify MFA details-Give a name, and select MFA device – authenticator app(if you want to choose mobile for authentication) and click on next.



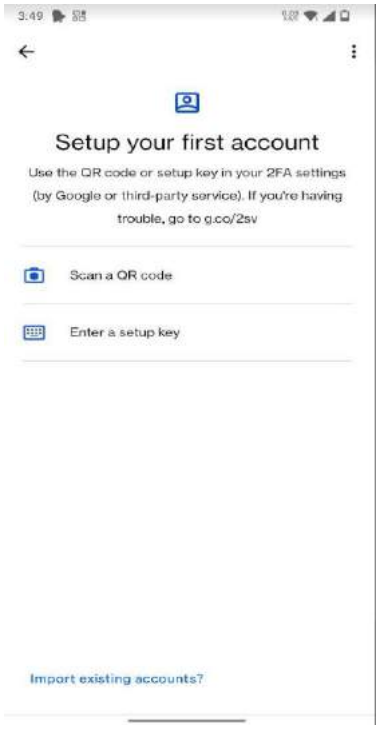
6)Next setup up your device by following the steps given below.



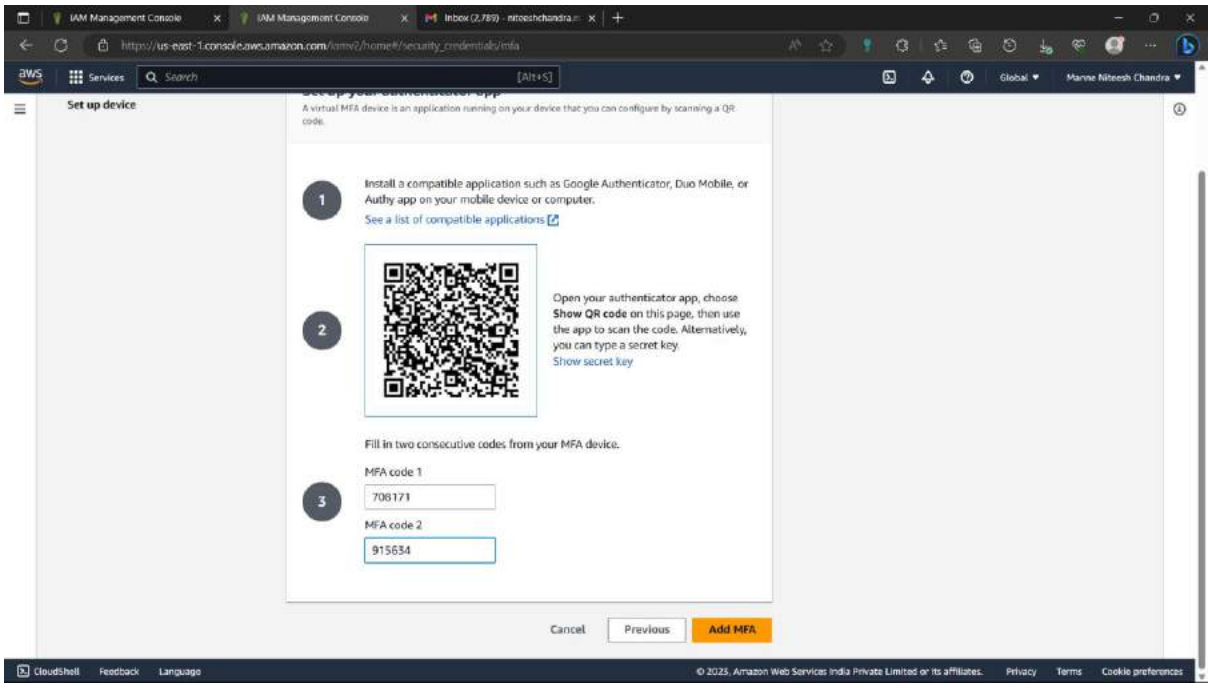
7)Now install Google Authenticator from play store to your device.



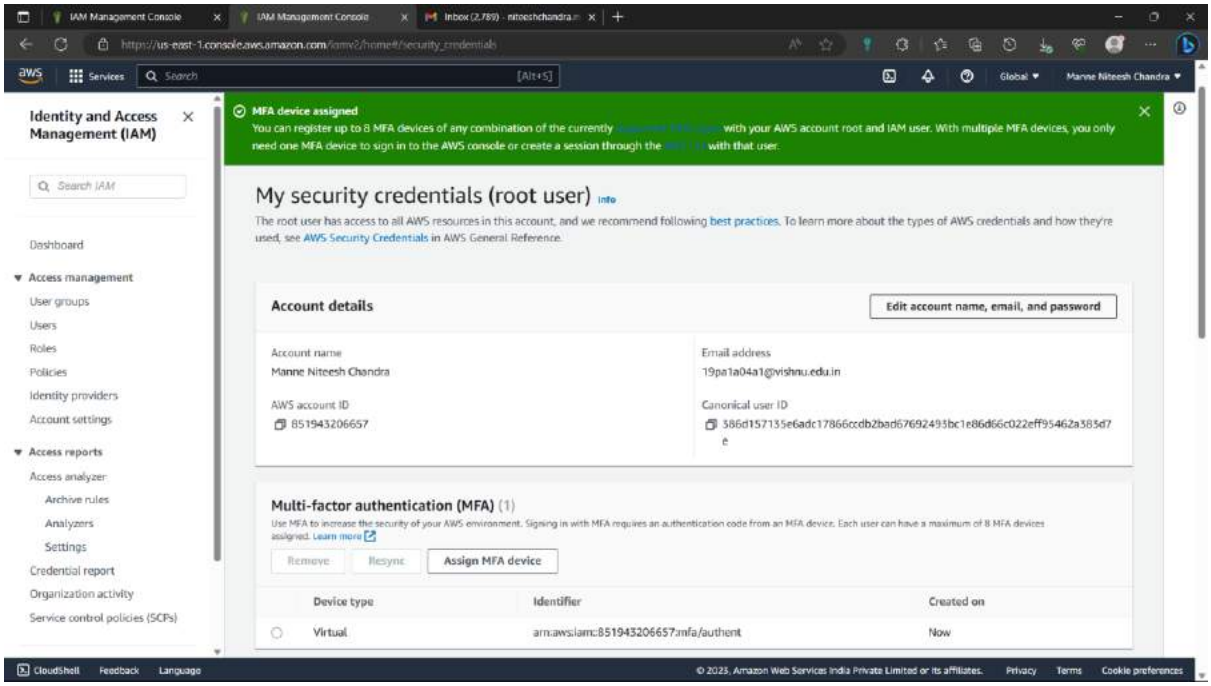
8)After logging in click on “Scan a QR code” scan the QR code which will showed at (point-2) by clicking “show QR code”.



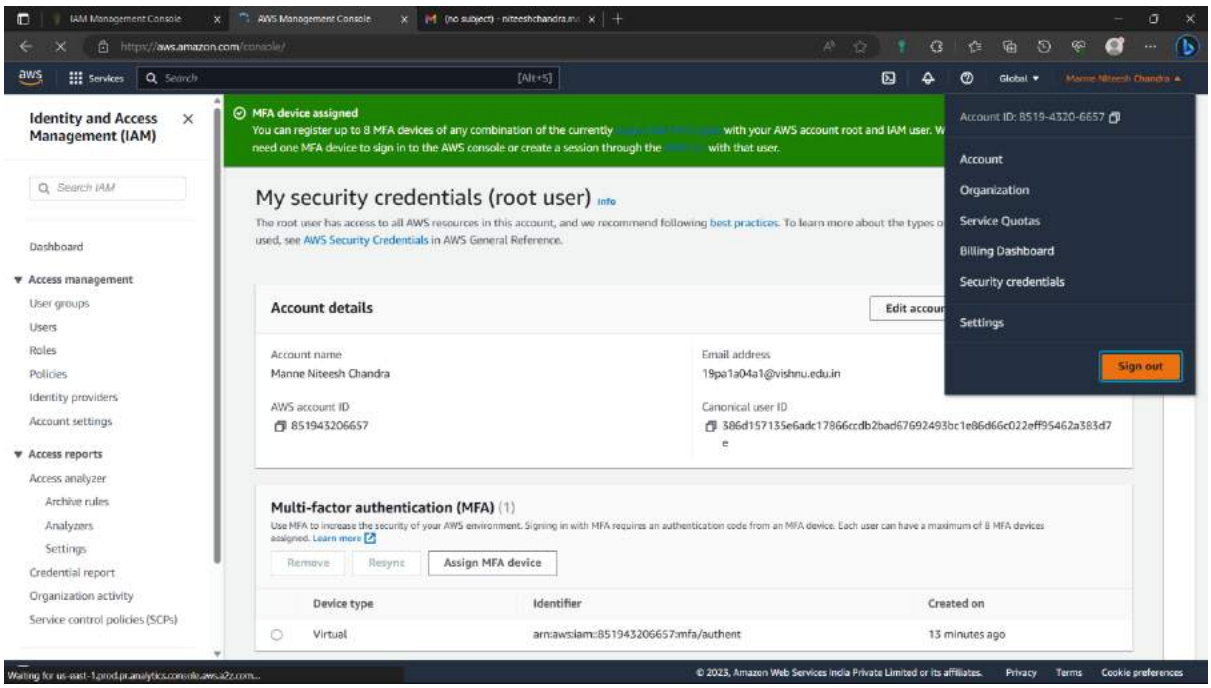
9)After scanning the QR codes you will get OTP in 6 digits, type that OTP in MFA code 1 after a few seconds you will get to another OTP type it in MFA code 2, click on “Add MFA”.



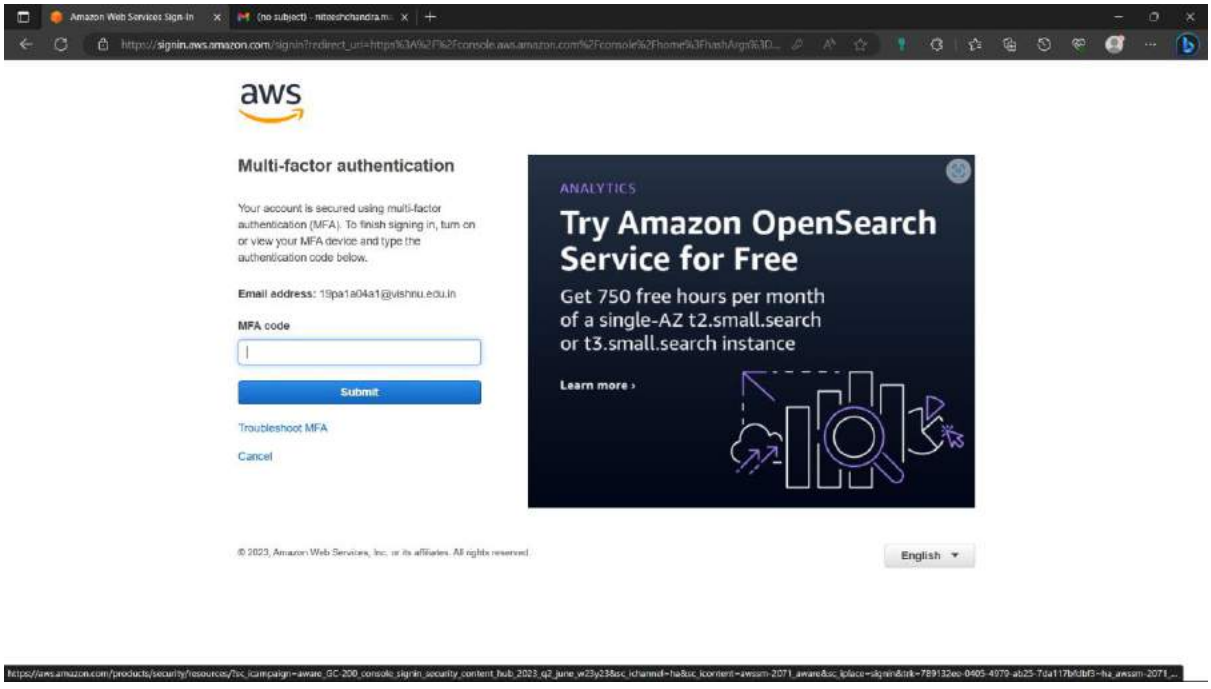
10)Now the MFA is created.



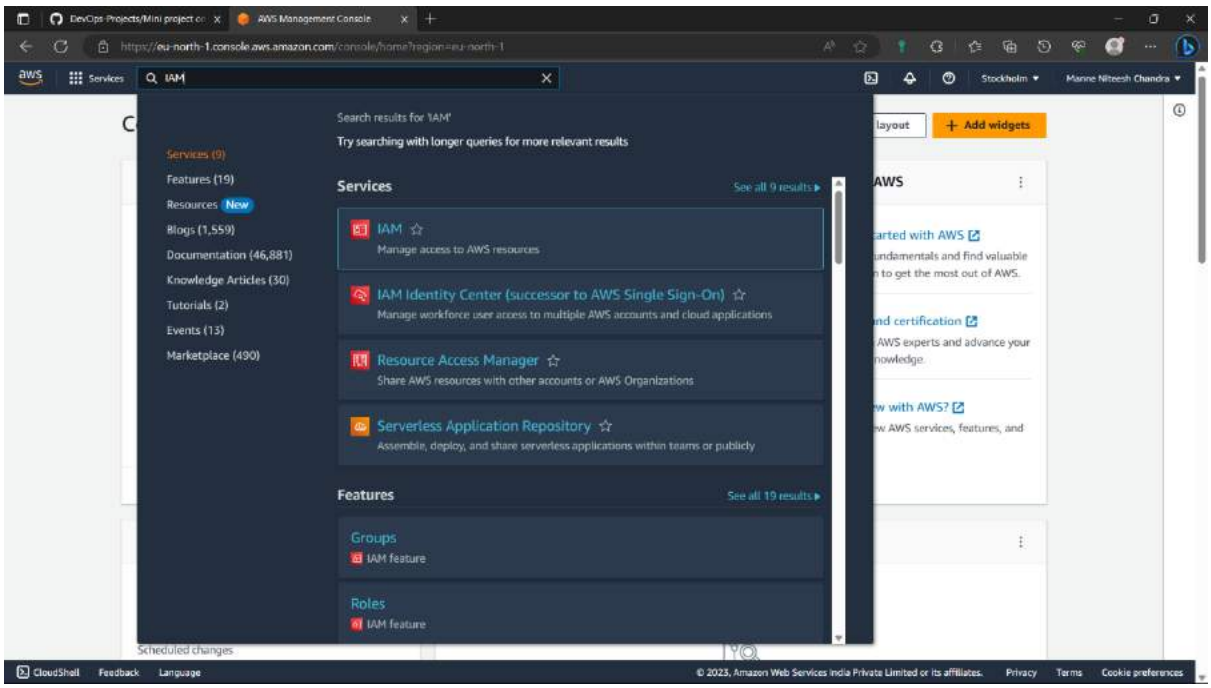
11)Now log out of your AWS console.



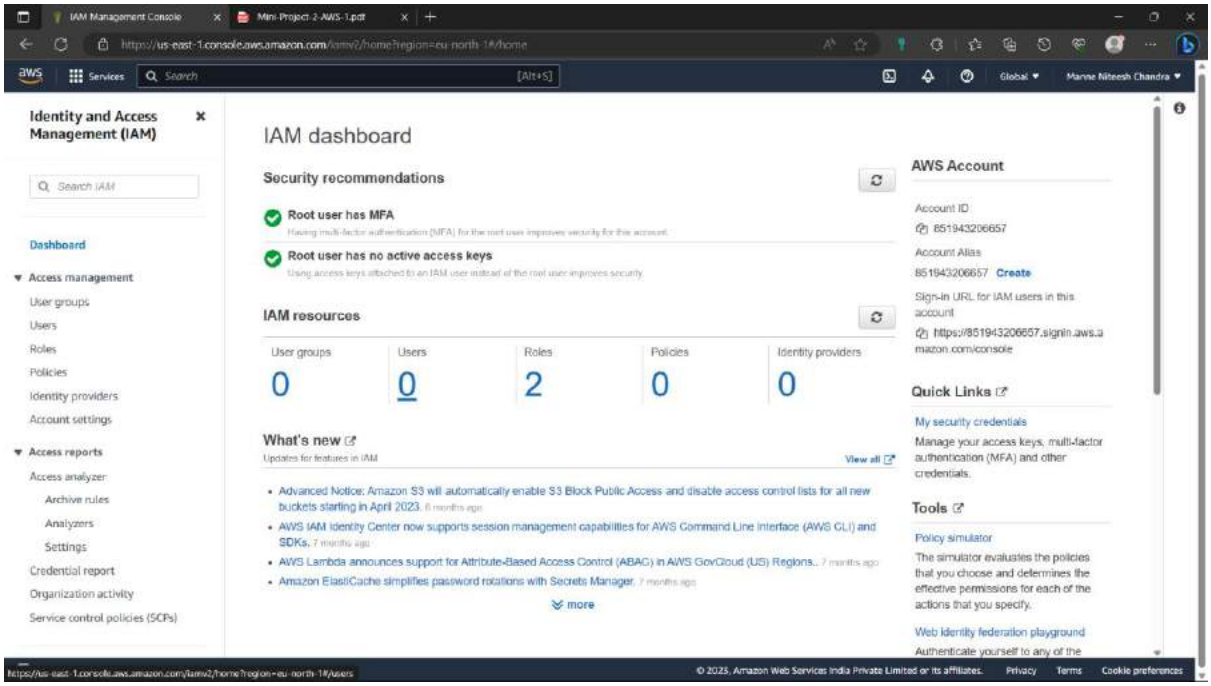
12)Now Login again. This time it will ask for MFA code, type the six digit code that will appear in google authenticator to login to your console.



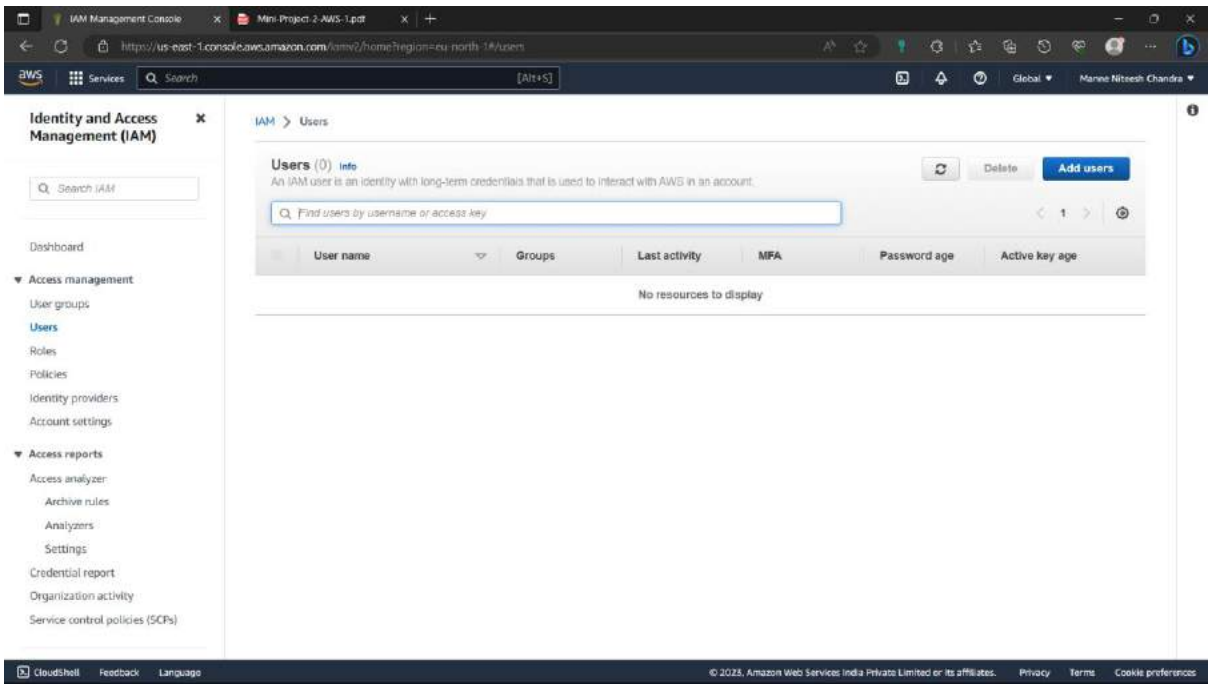
13)Go to IAM in AWS console.



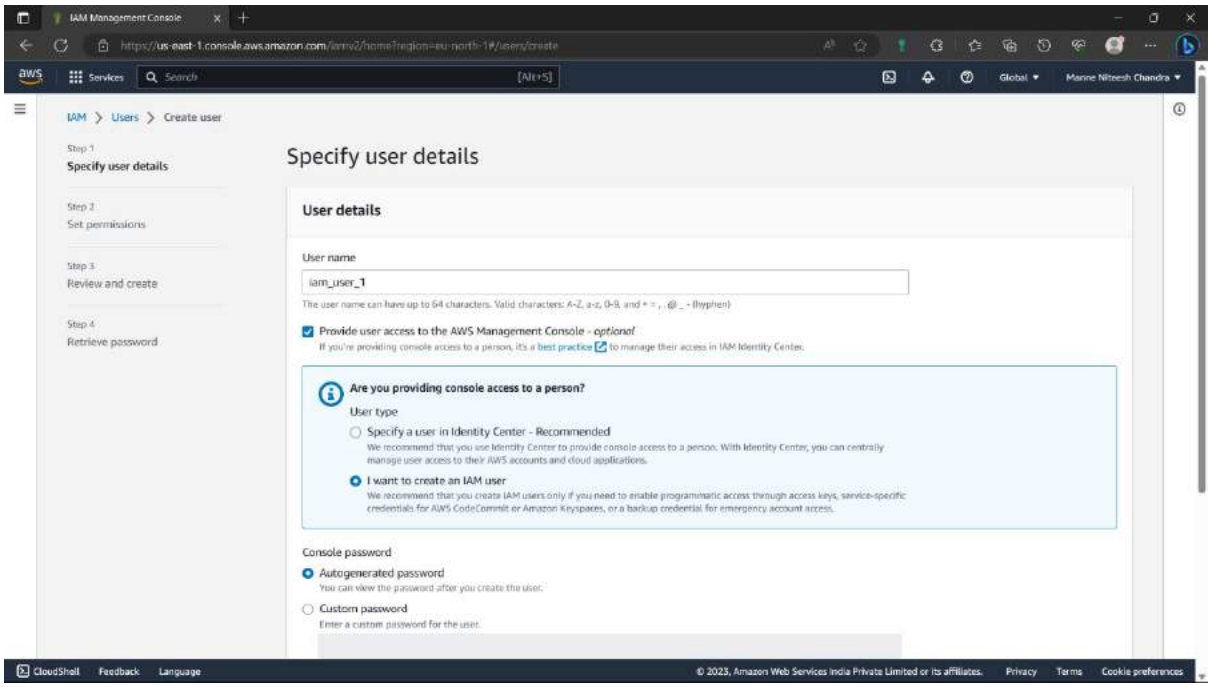
14)Click on users.



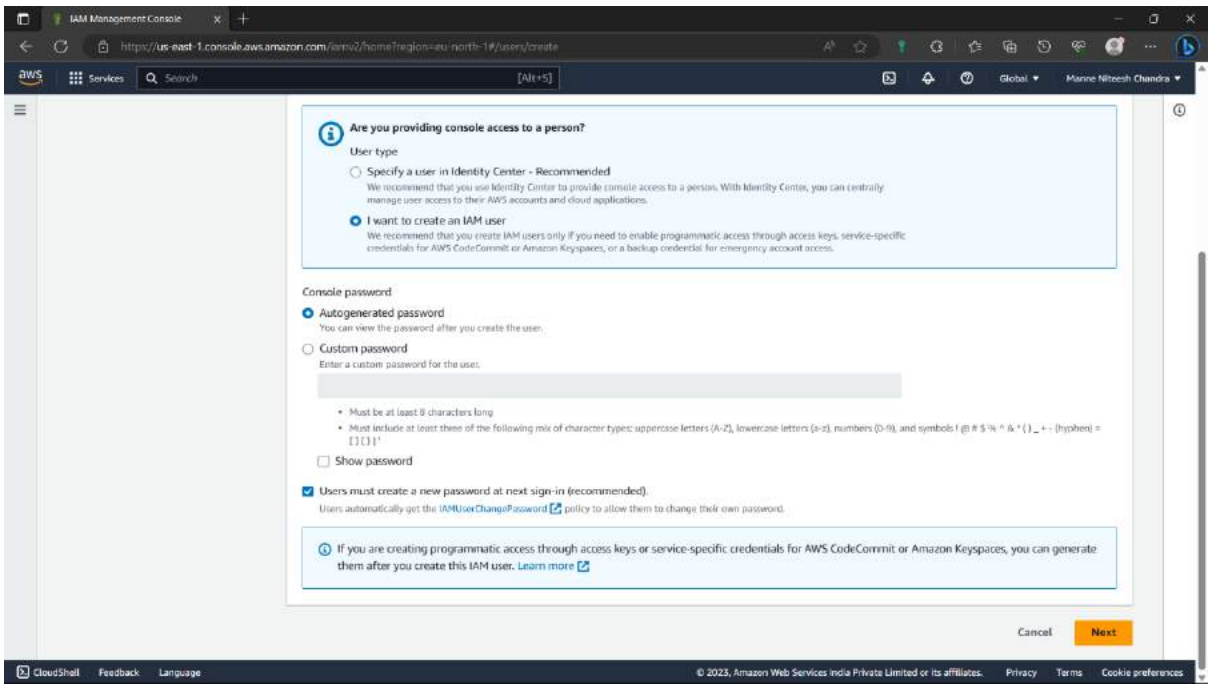
15)Click on Add user.



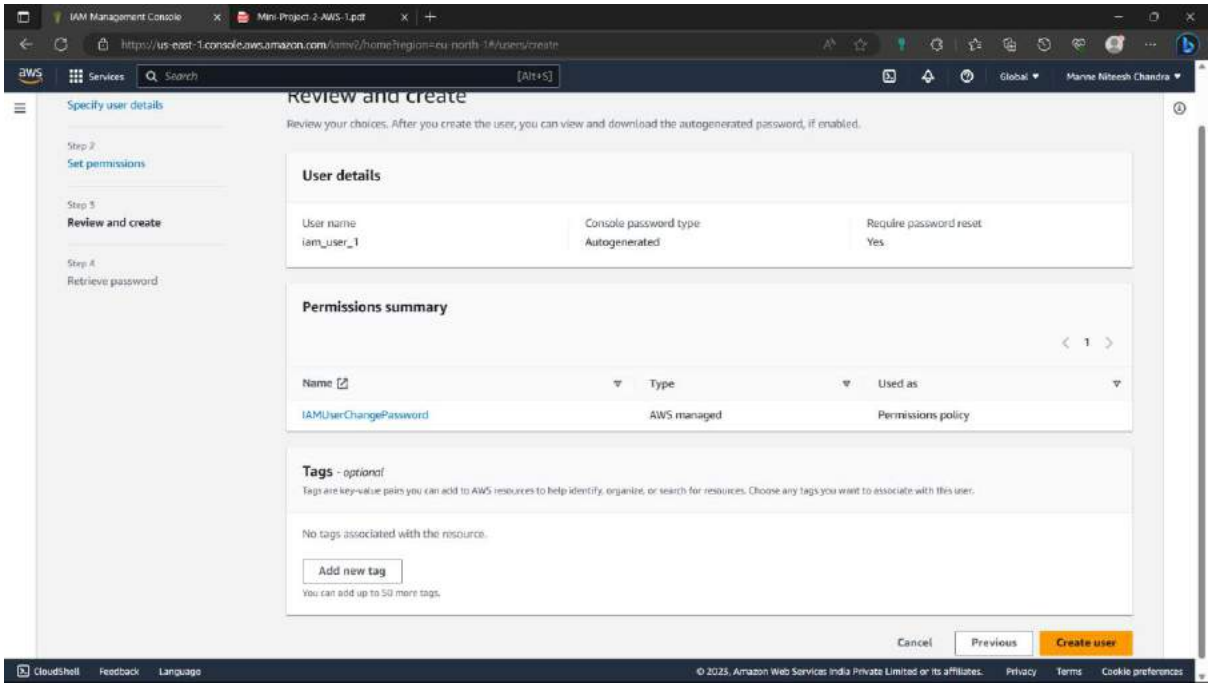
16)Give the name of your choice, enable user access, enable I want to create an IAM user, enable Autogenerated password.



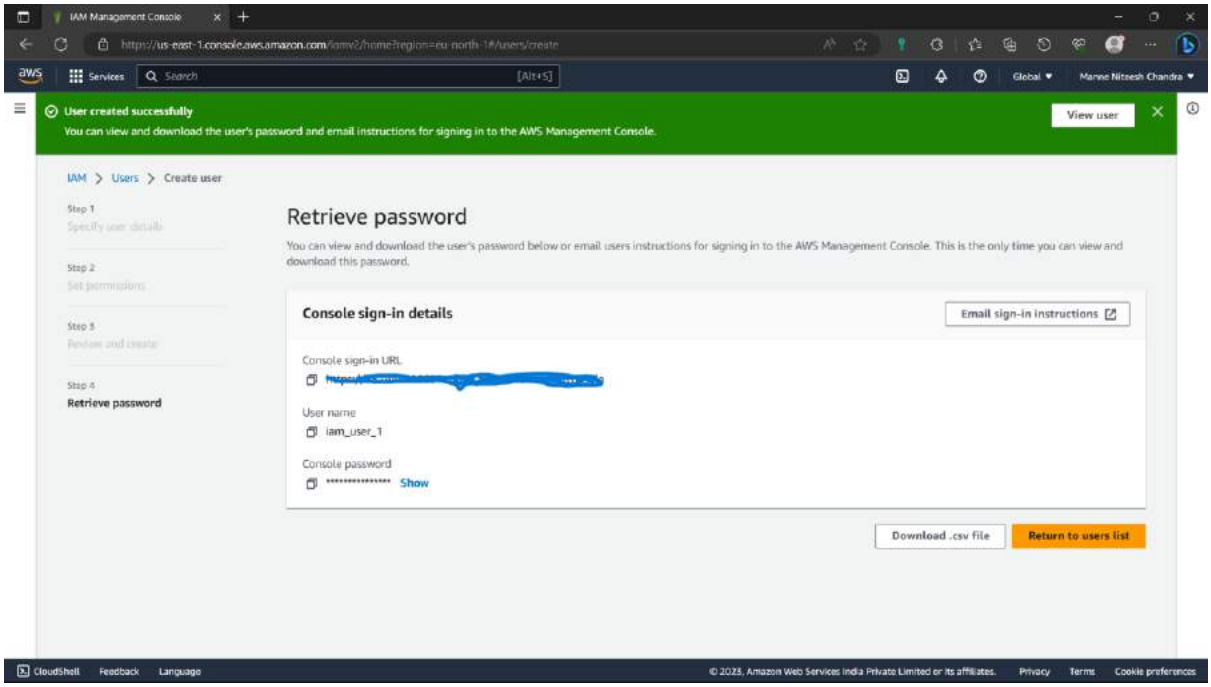
17)Enable “User must create for the next sign-in” and click on next.



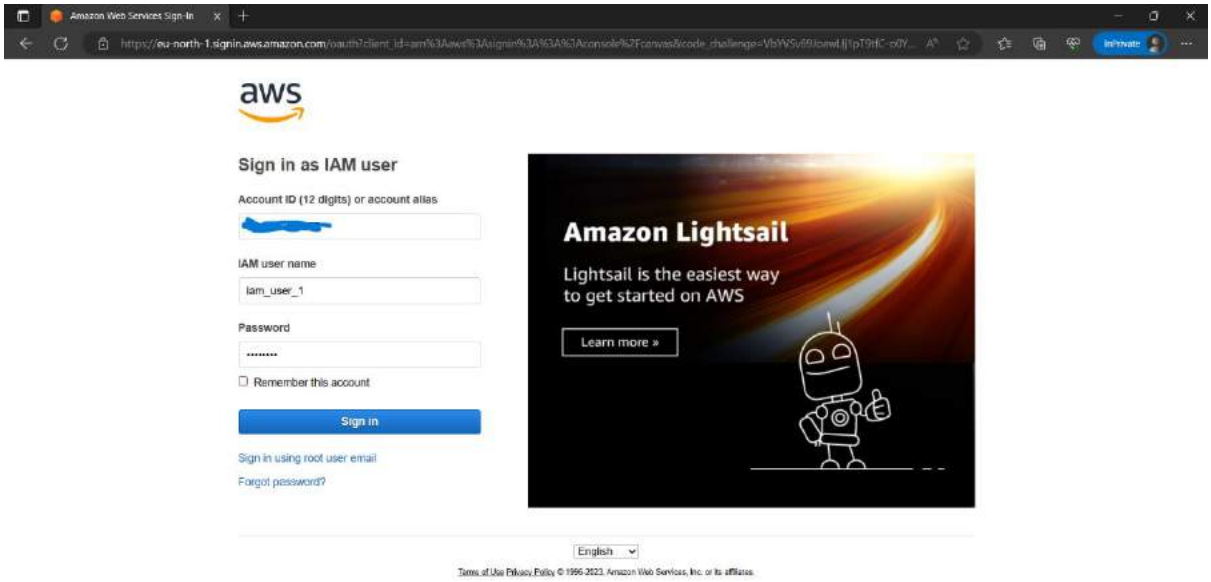
18)Review all the details and click next.



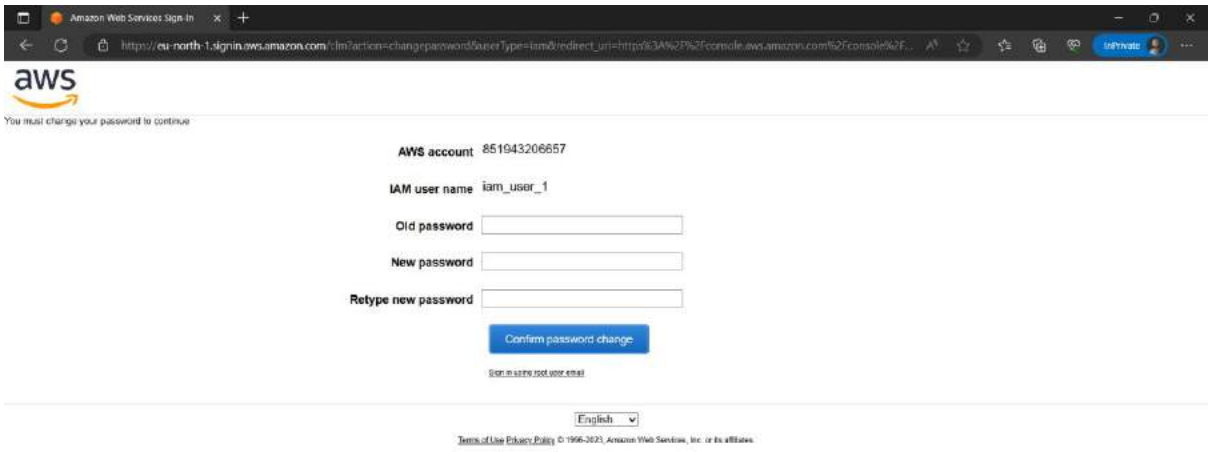
19)Now the IAM user is created. Copy the details save it file(Excel, word, notepad).



20)After saving copy the sign-in url and paste it in a incognito tab(ctrl+shift+n). By default there will be account id, type the username and password and click sign in.

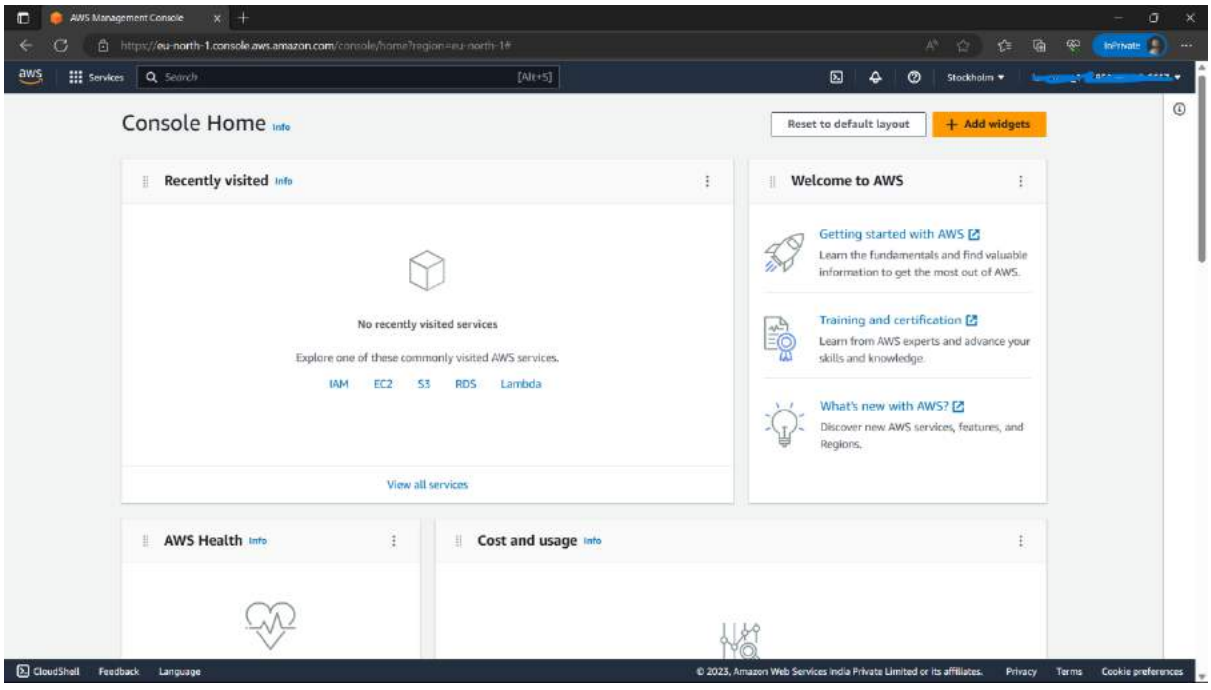


21)Now it will ask to change the password, let IAM user change the password so that they can remember it easily. After typing the password click on “confirm password change.”

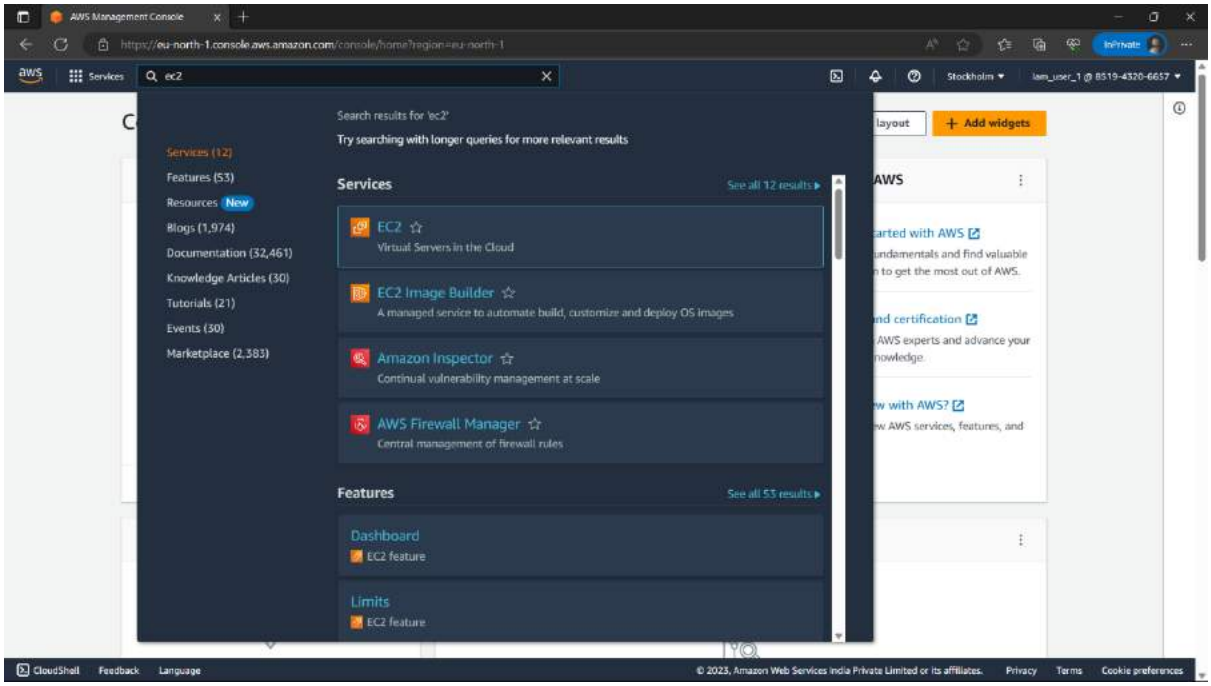


---

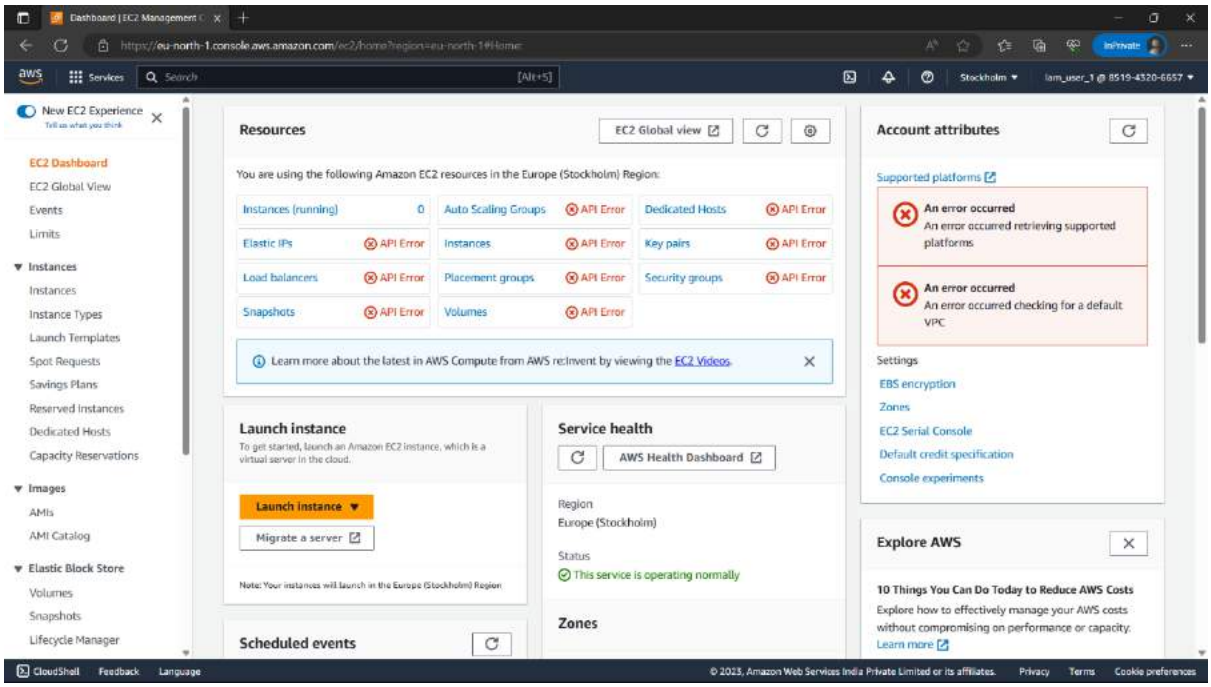
22)Now we are signed in as a IAM user.



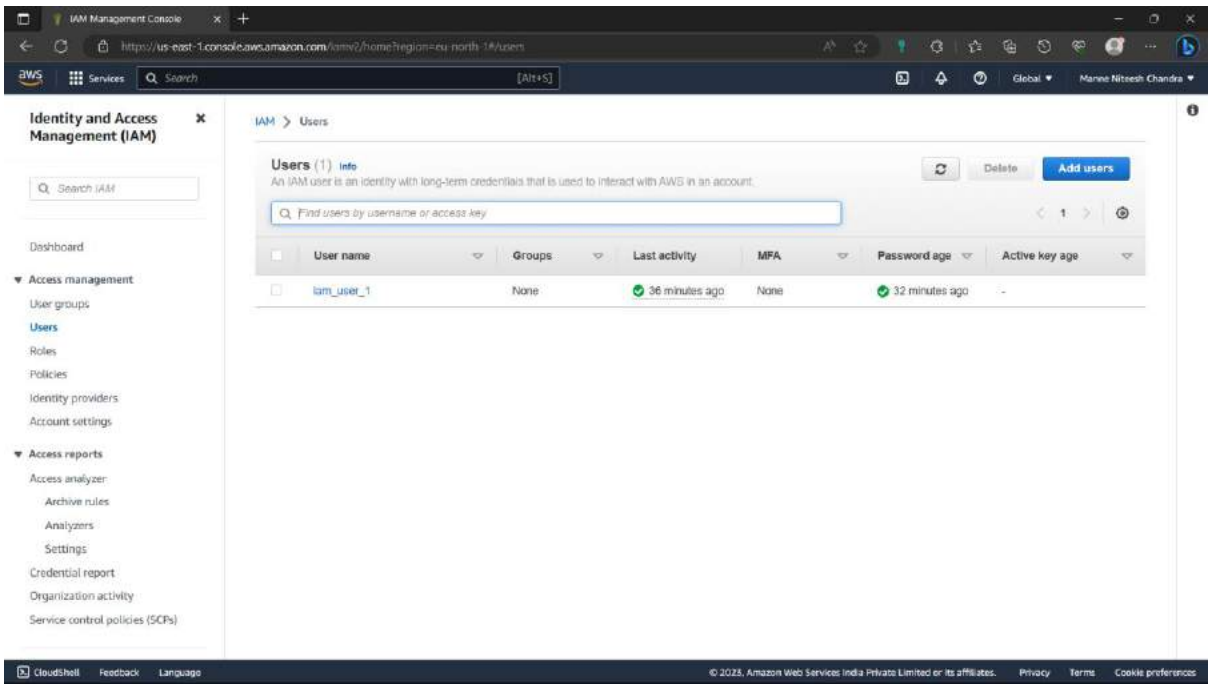
23)Now open ec2 by searching in the search bar click on ec2 instance option.



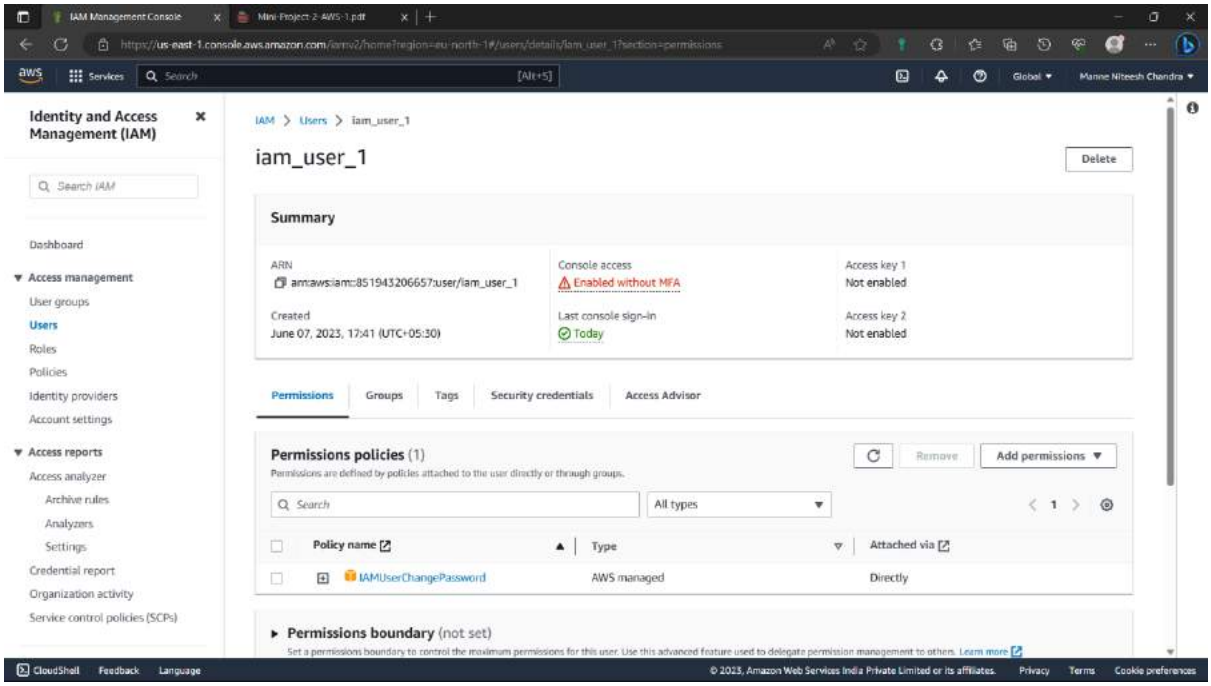
24)We see that it shows API error because we don't have ec2 permission to create an instance.



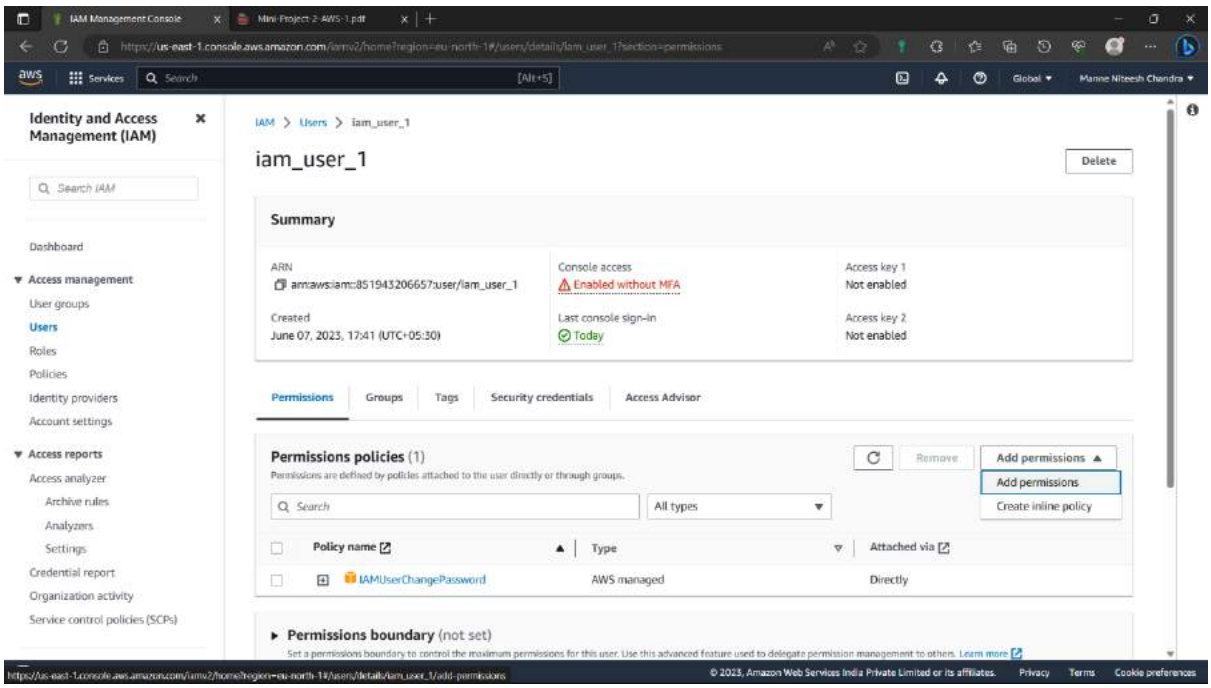
25) Now go to IAM user in root account which created the IAM user.



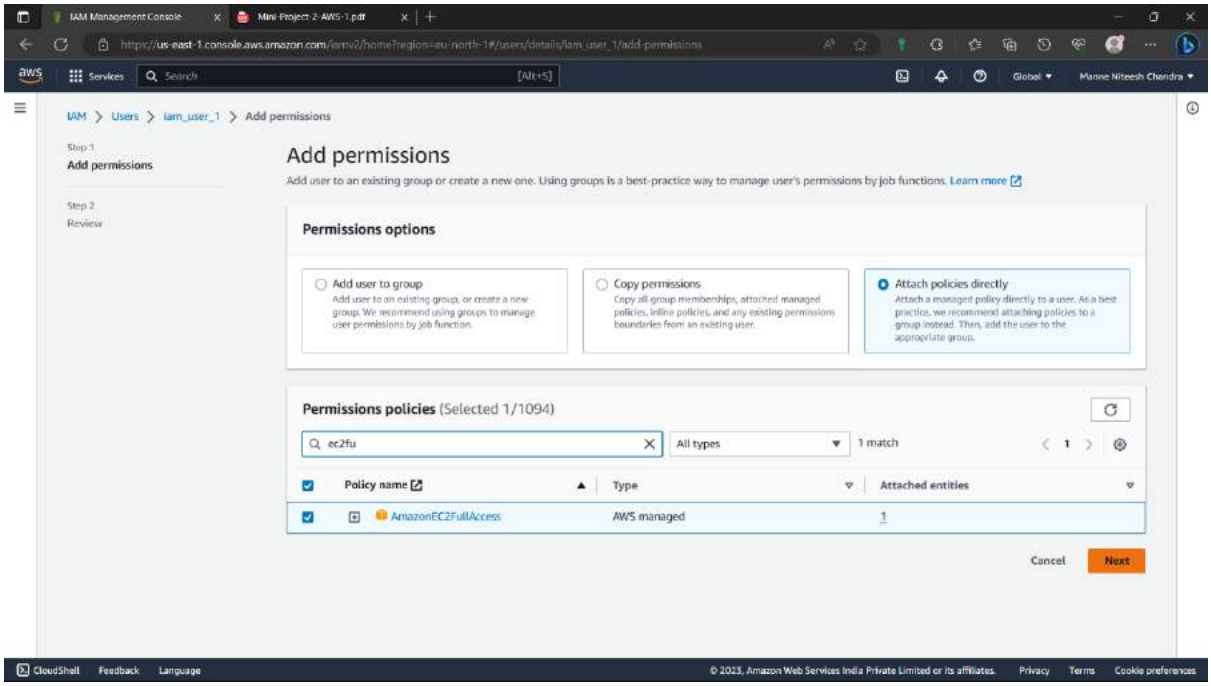
26) Click on the created IAM user.



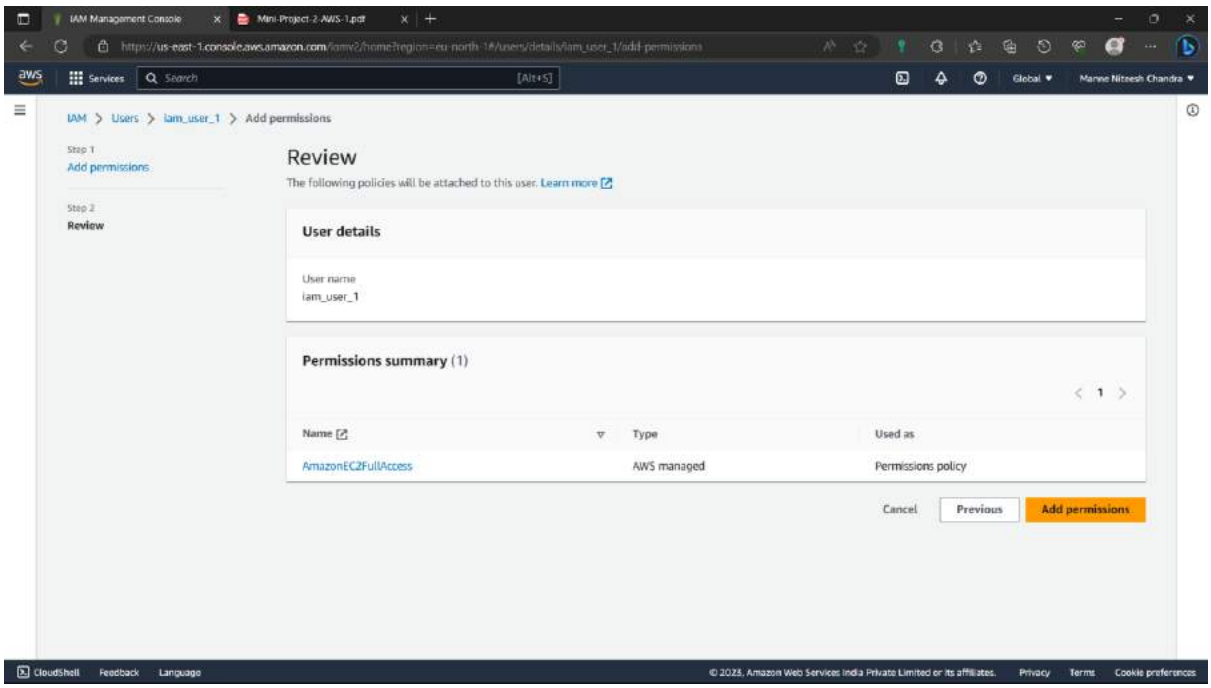
27) Click on add permission option.



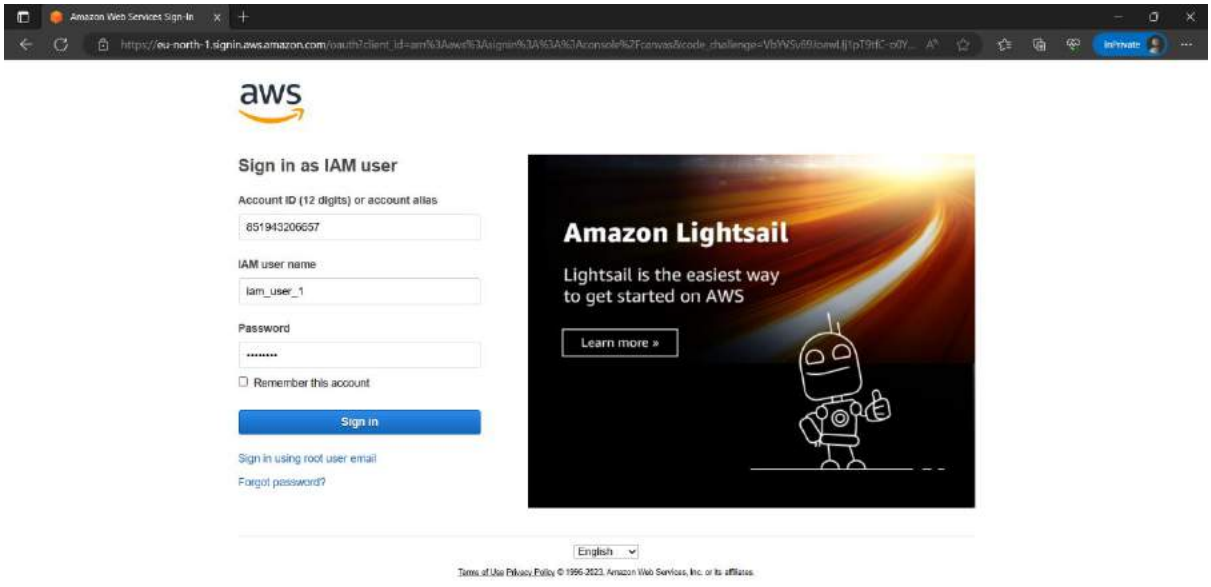
28)Select “Attach policies directly” option and select “AmazonEC2FullAccess” permission and click on next.



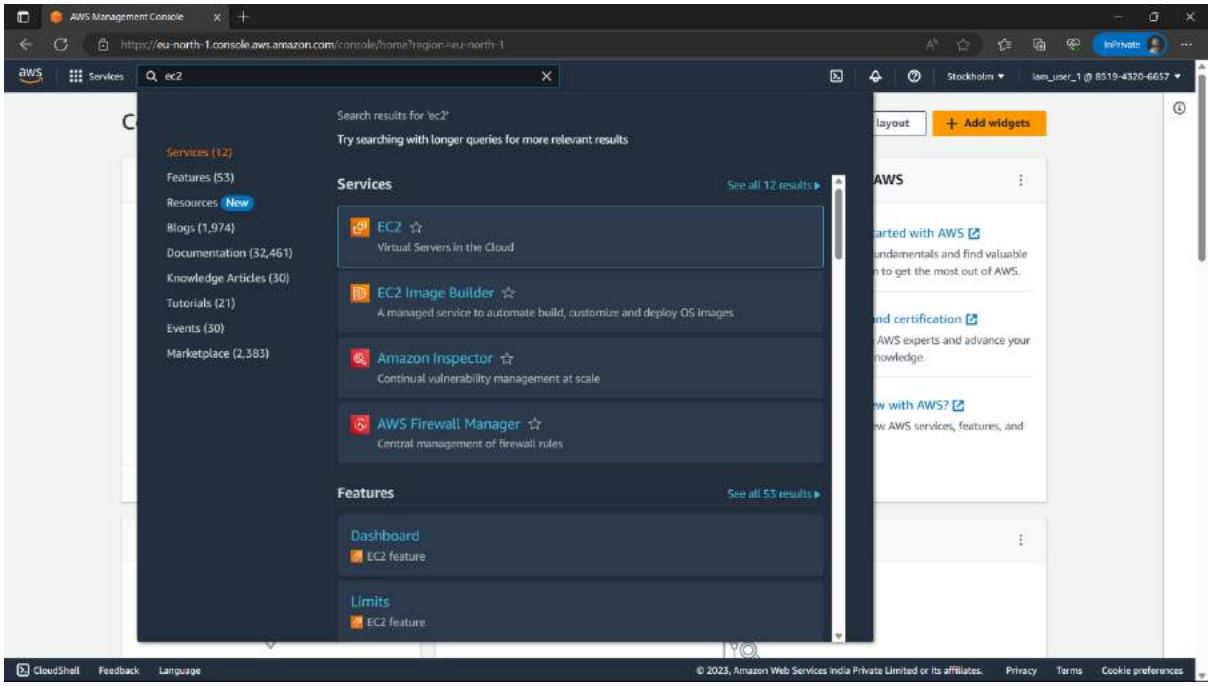
29)Review the assigned data and click on Add permission.



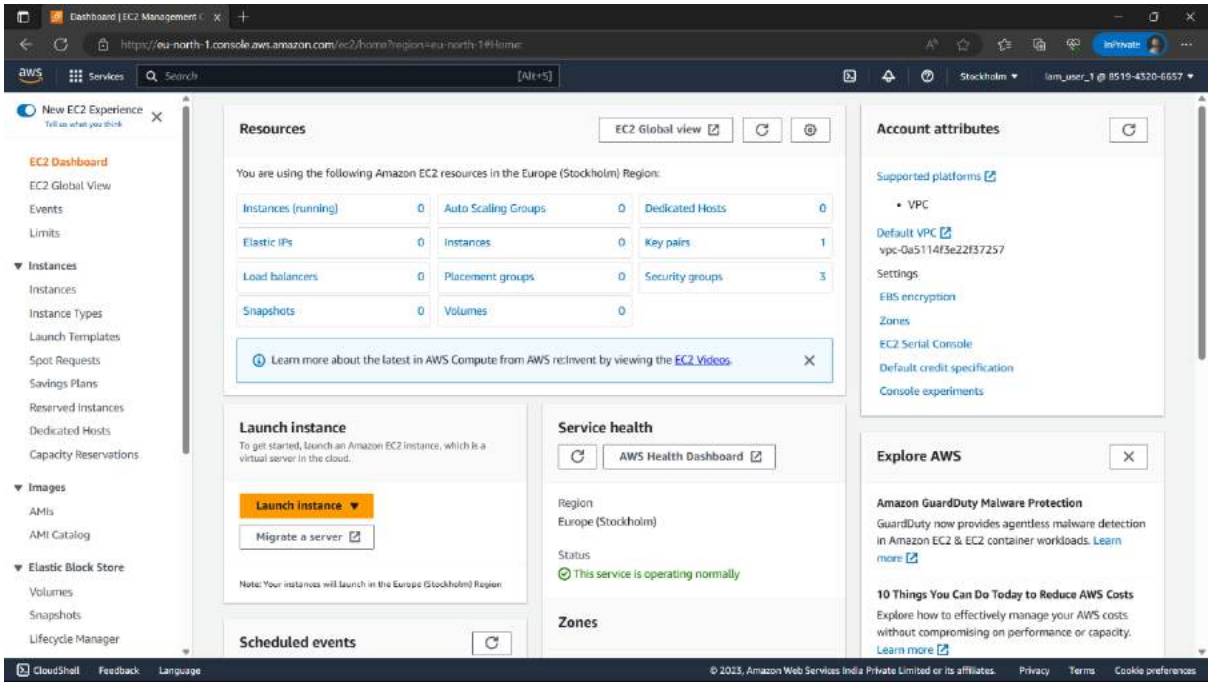
30)Now open incognito tab and sign in as IAM user.



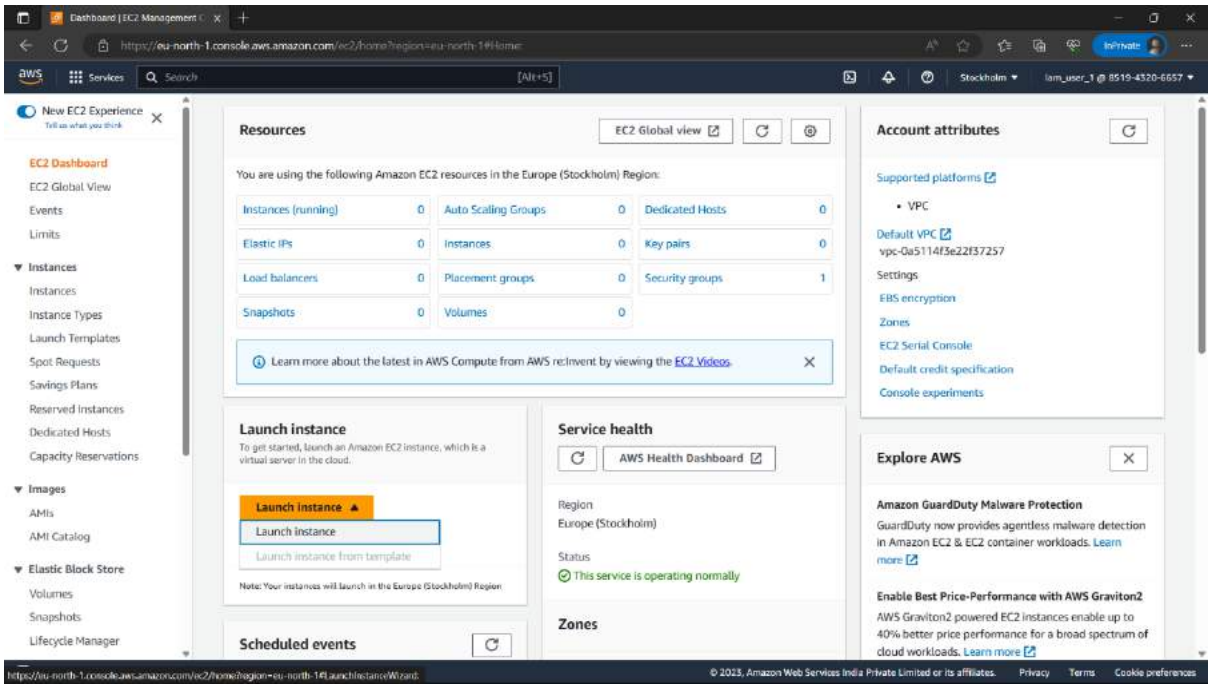
31) Now open ec2 by searching in the search bar click on ec2 instance option.



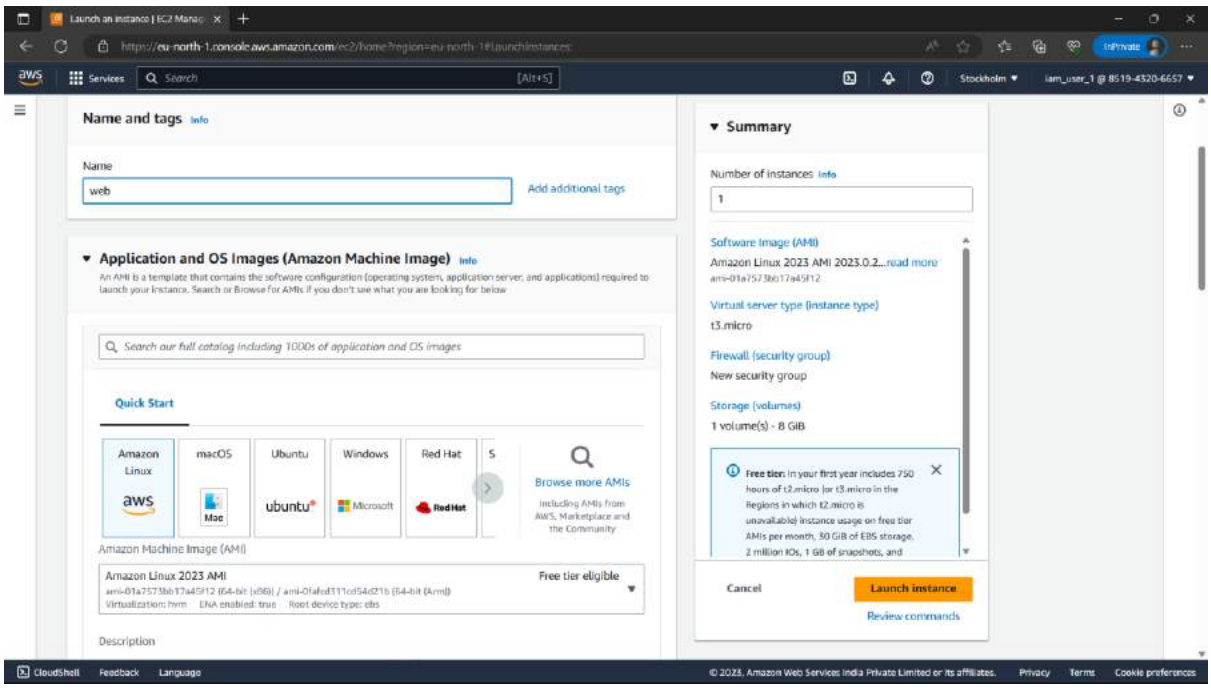
32)Now we don’t see any api error because we added “amazonec2fullacces” permission.



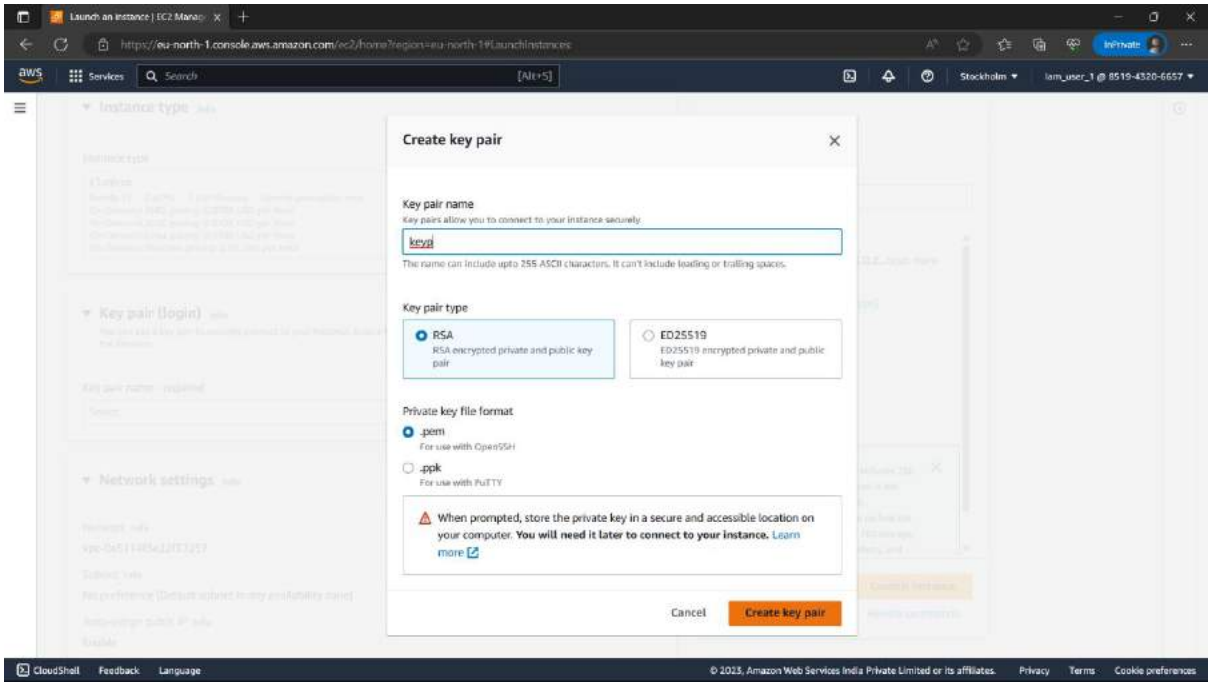
33)Now create an instance by clicking on launching instance.



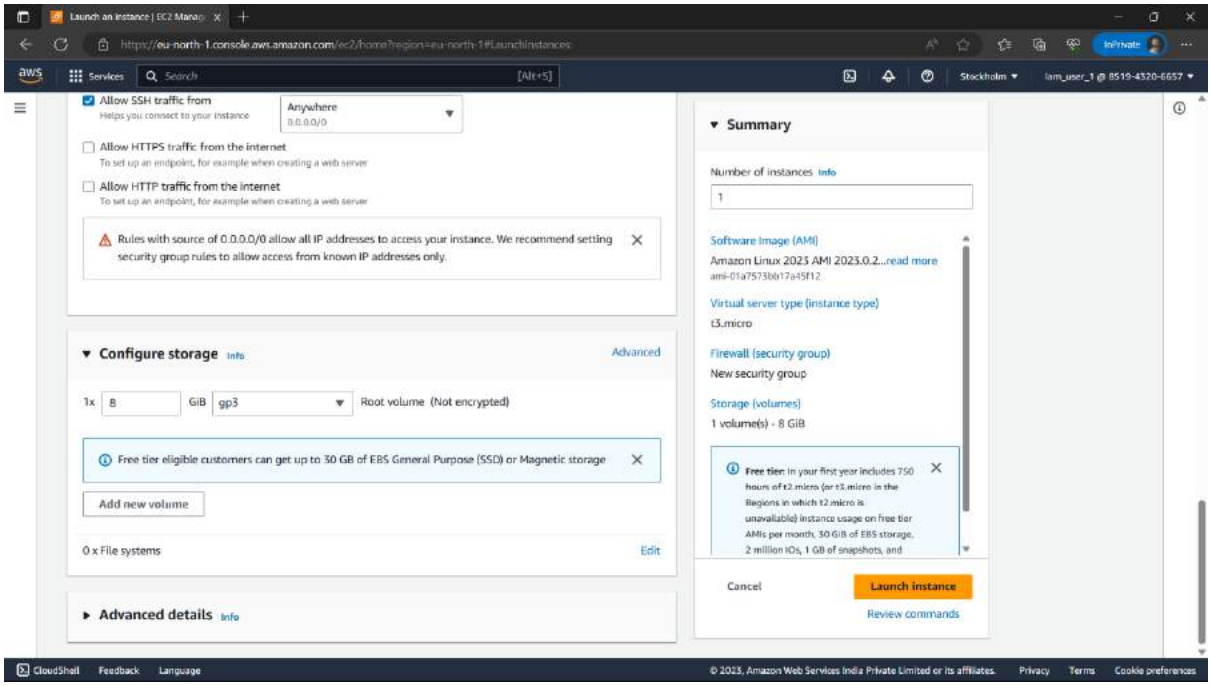
34)Give a name to the instance and let everything be default.



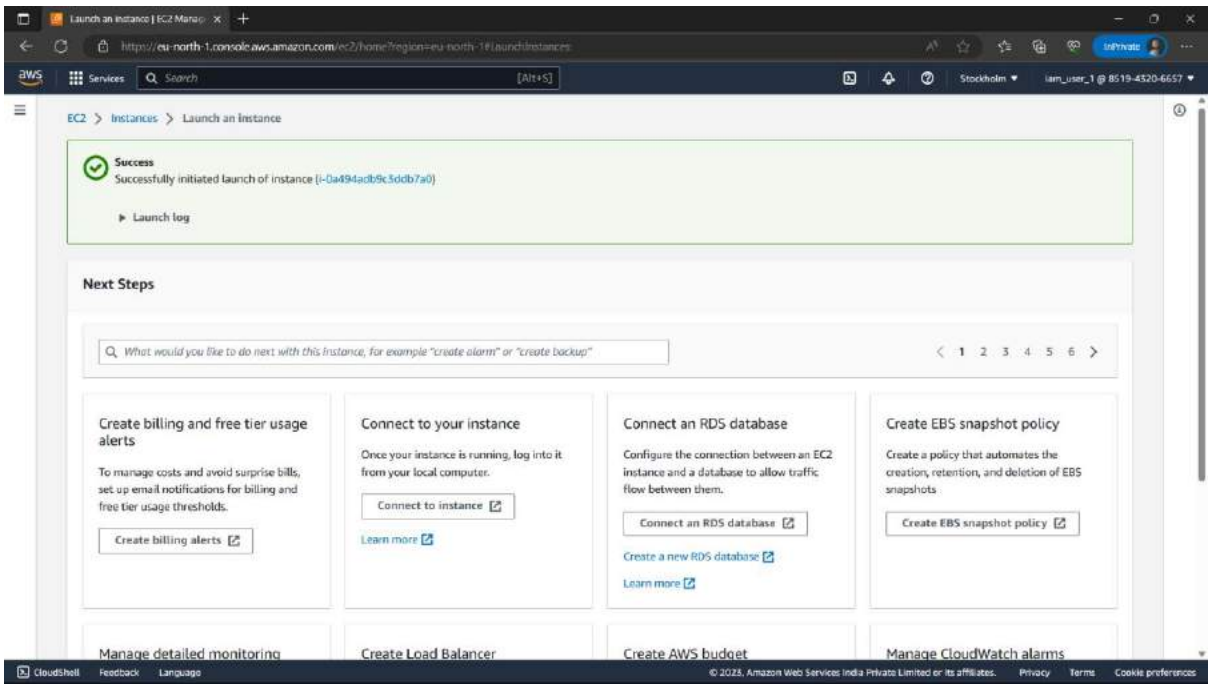
35)click on “create a keypair” and give a random name,type,file format and click “create key pair”.



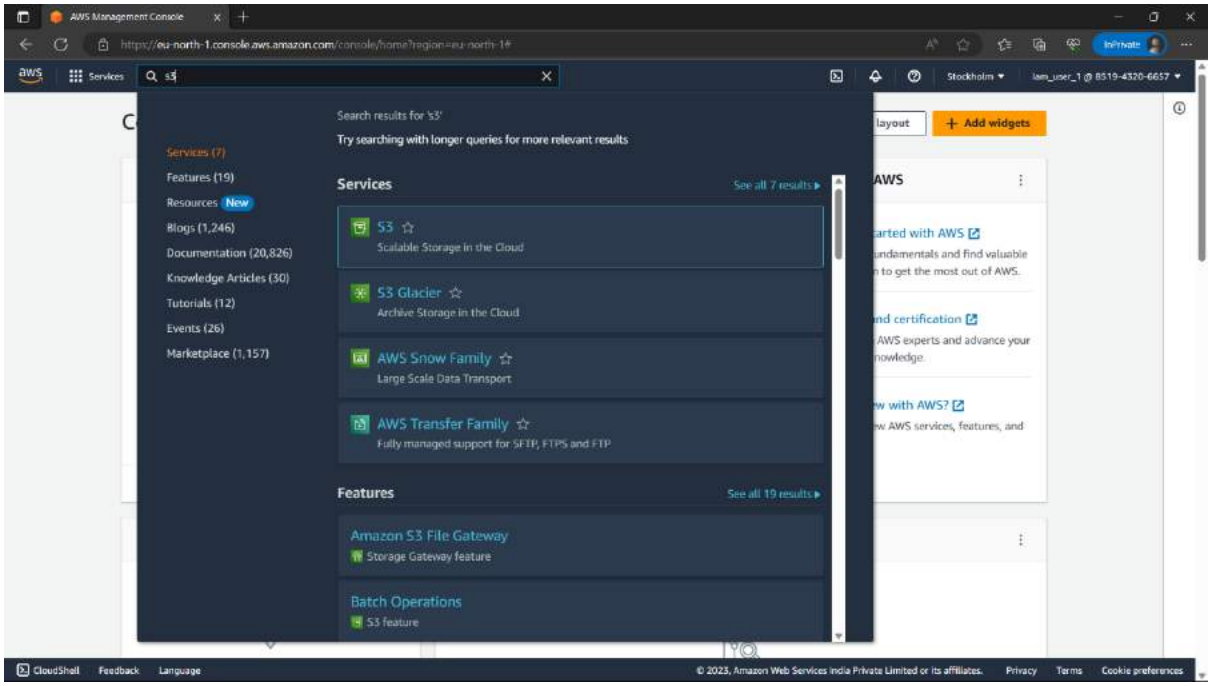
36)Review everything at Summary and click on launch instance.



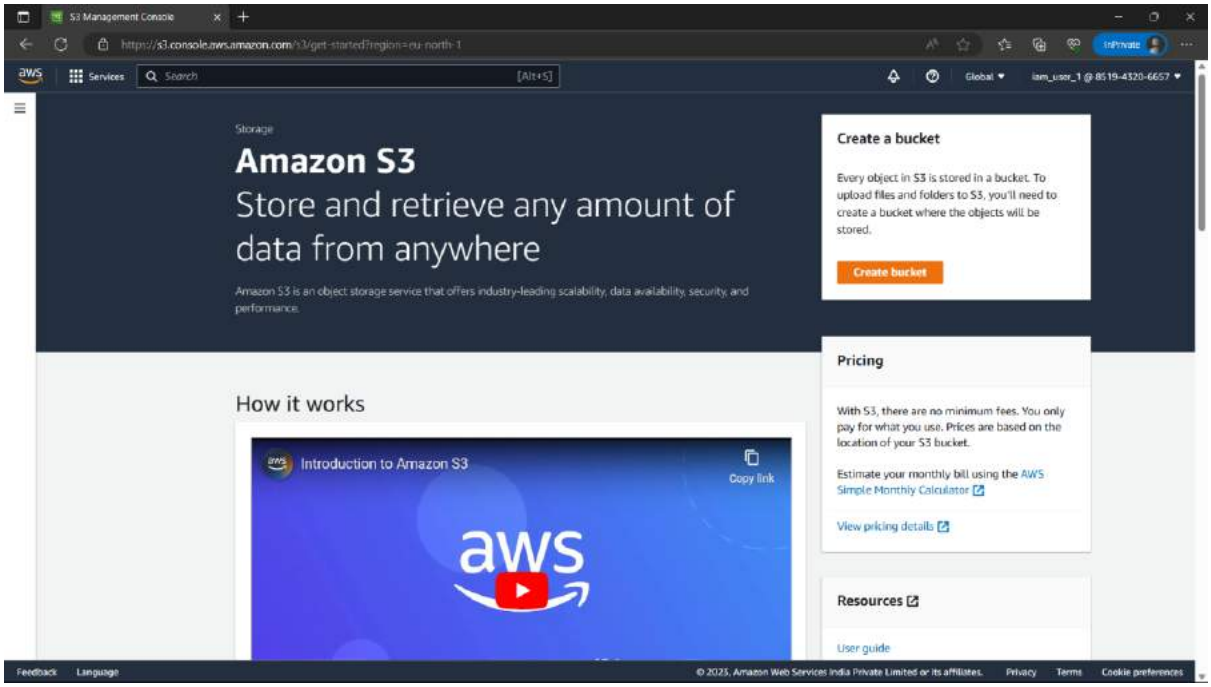
37)Now the instance is created.



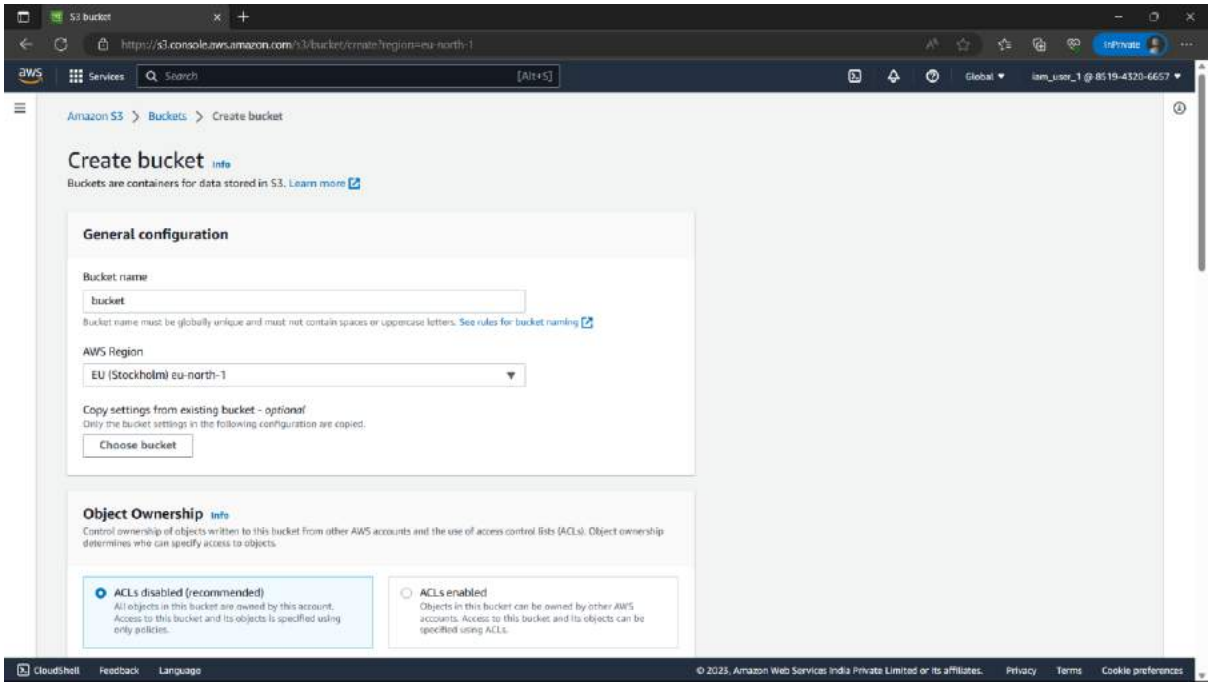
38)We created ec2 instance because we assigned “ec2fullaccess” permission to the “IAM” user we cannot create any other tool with this permission. Now let’s try to create a S3 bucket. In search bar type S3 and click on S3 option.



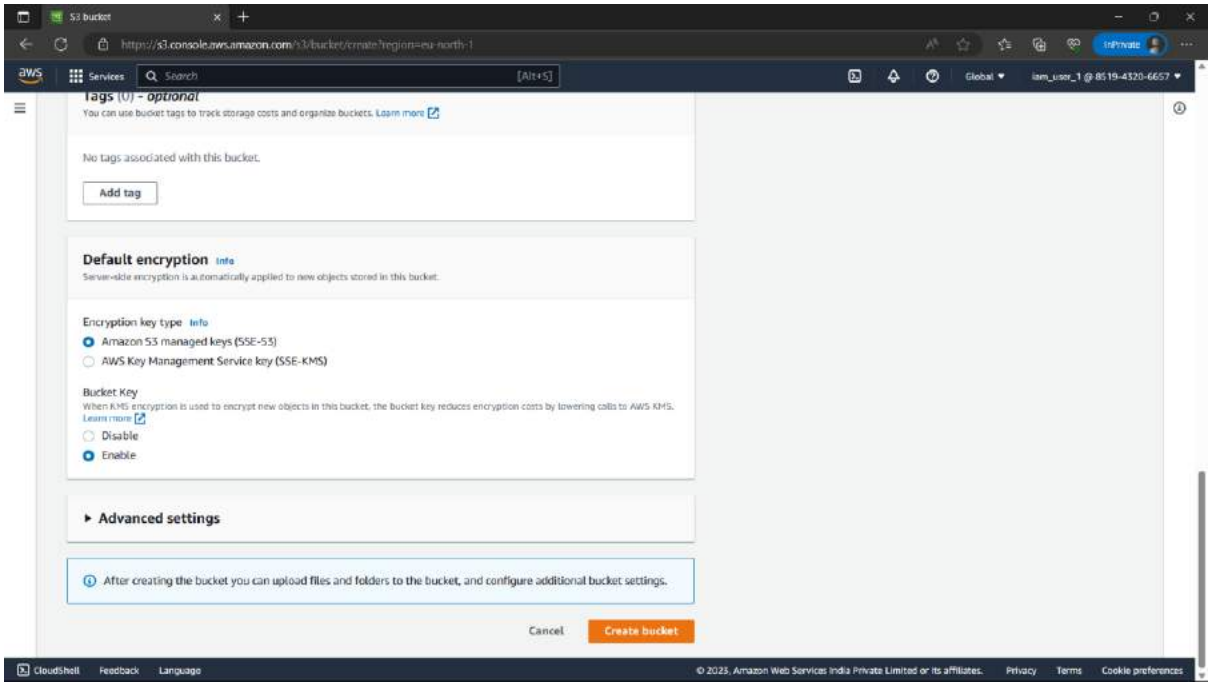
39)Click on create bucket option.



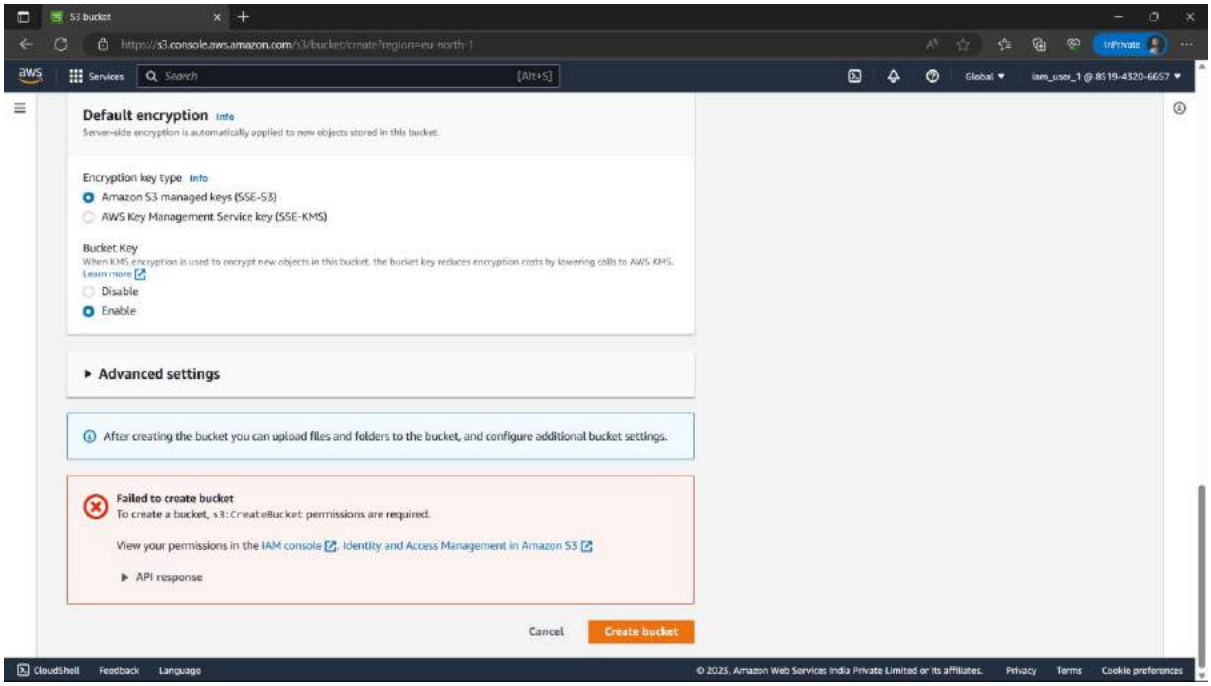
40)Now give a name to the bucket it has to be unique.



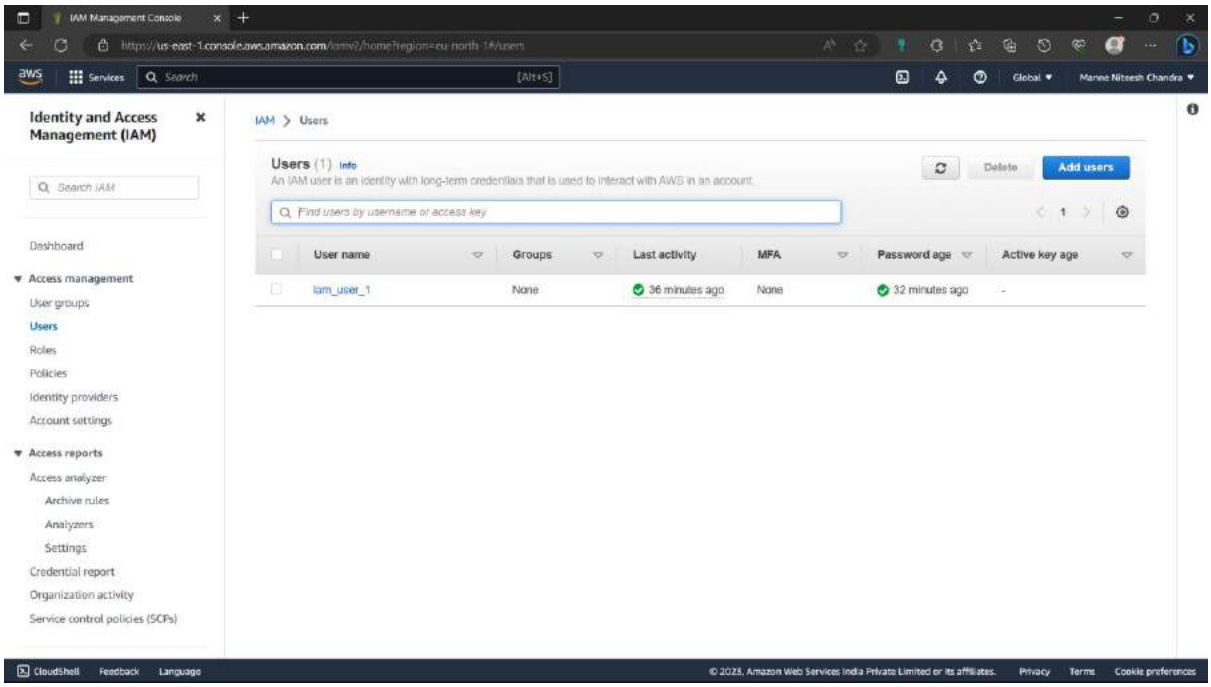
41)Let everything be default and click on “create bucket”.



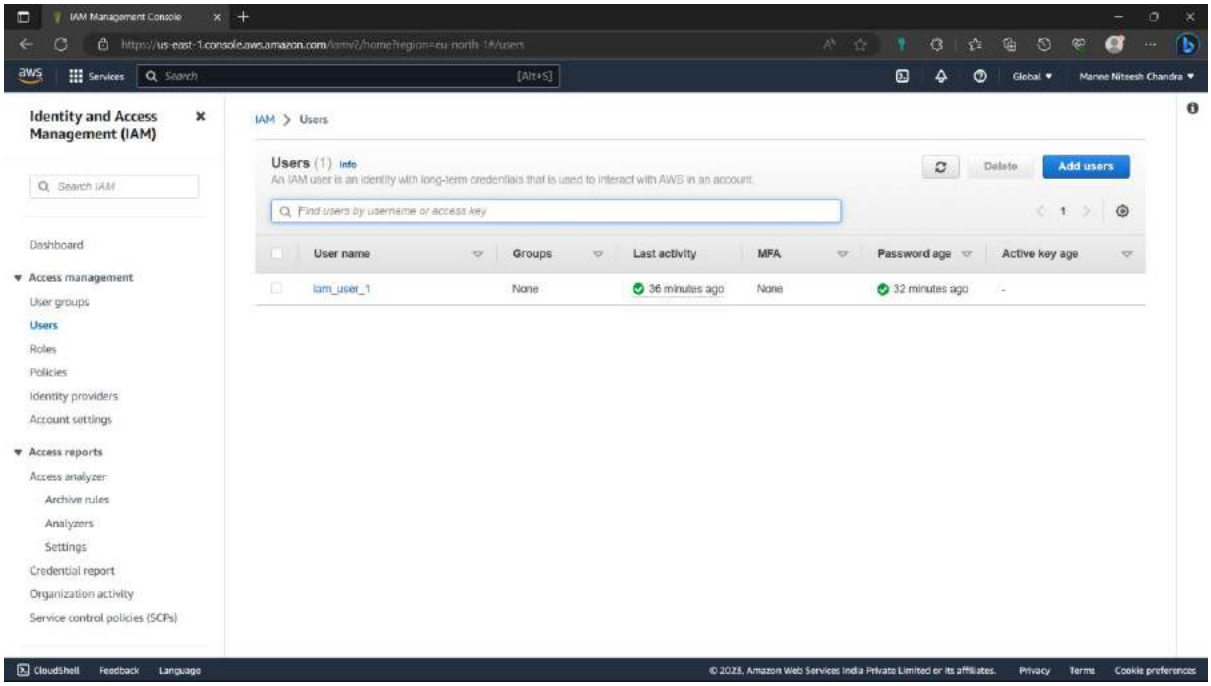
42)It will show that you do not have permission to create a S3 bucket. Now let's try adding administrator permission to the user. with administrator we can create anything.



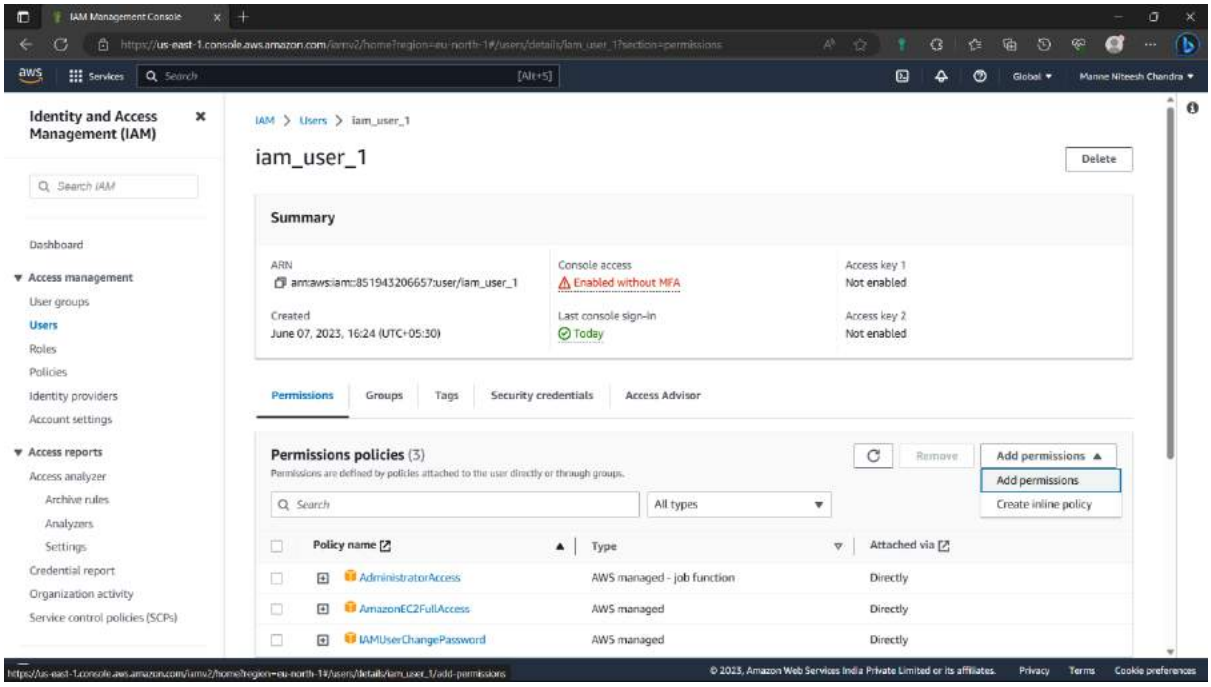
43)Now go to IAM user in root account which created the IAM user.



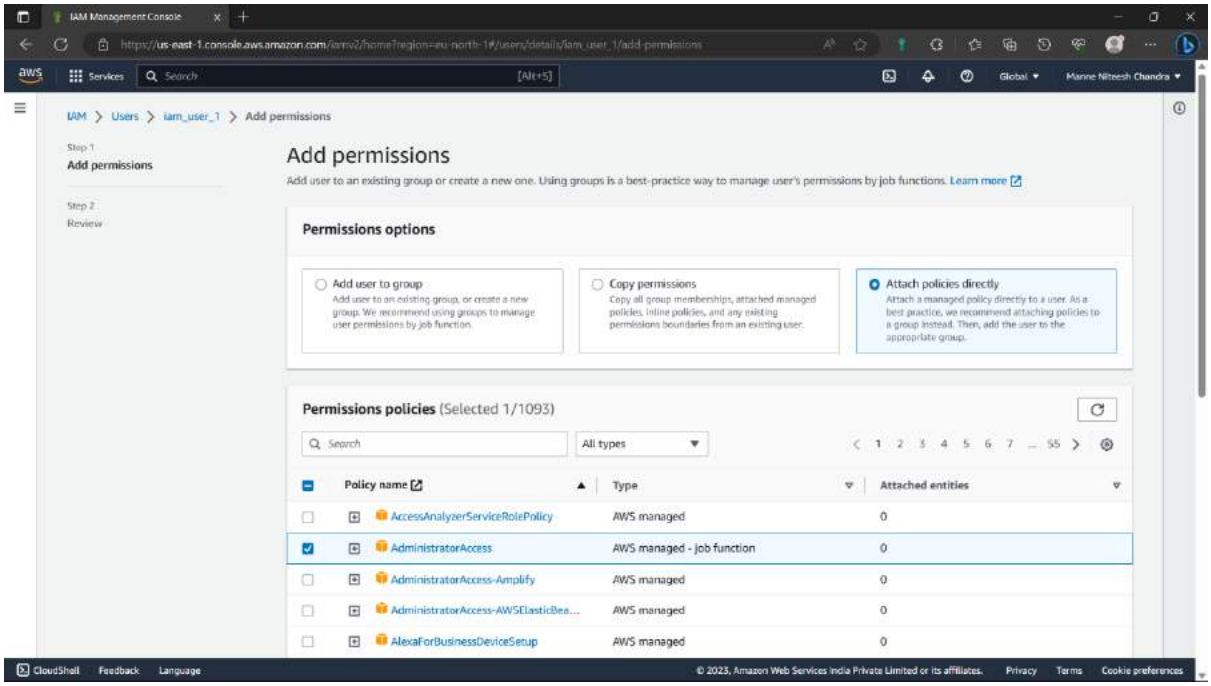
44)Click on the created IAM user.



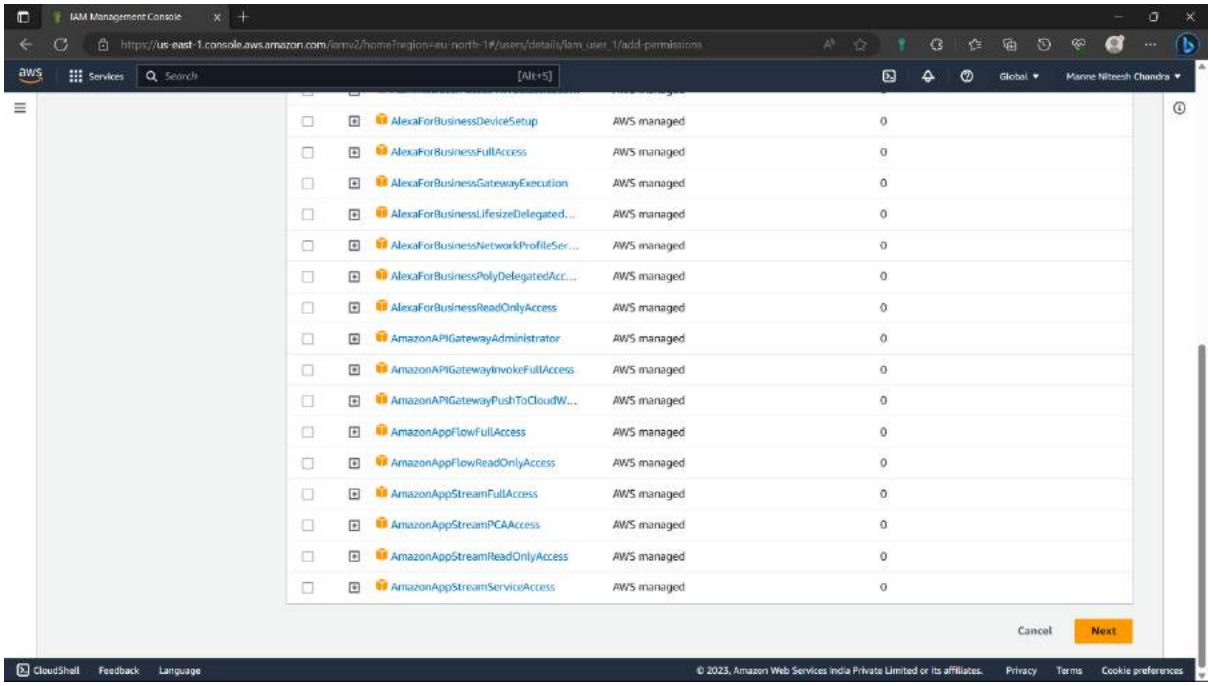
45)Click on add permission option.



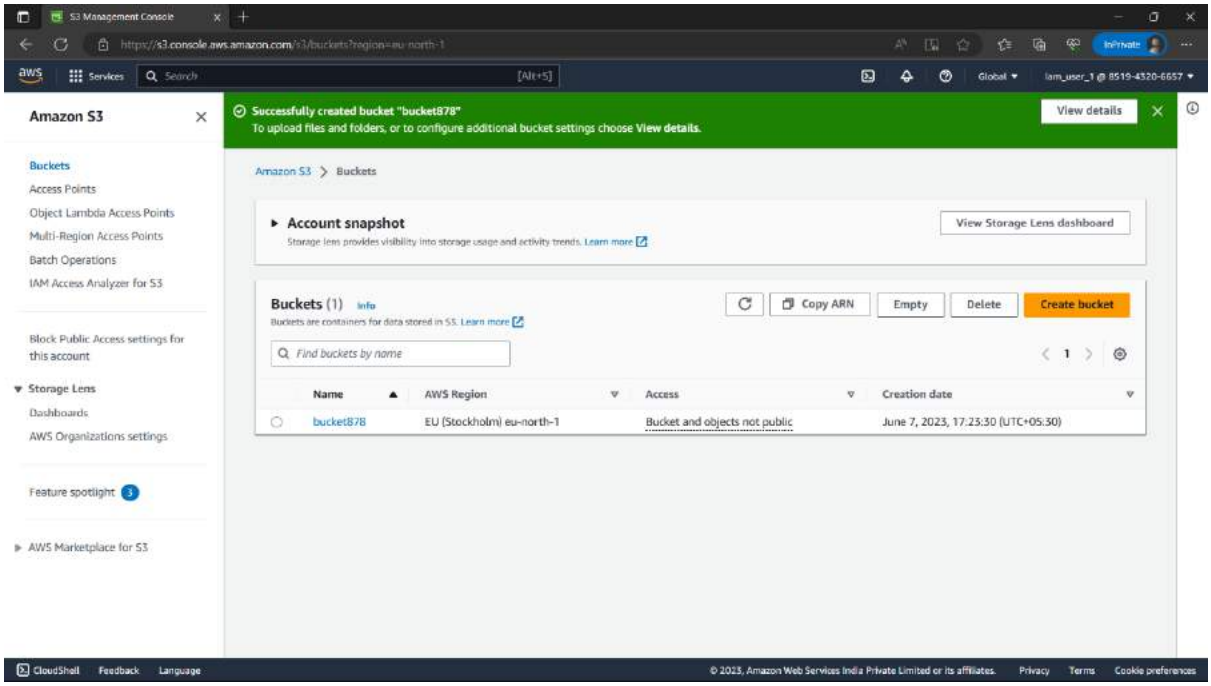
46)Type “administratoraccess” and select it.



47)Click Next.

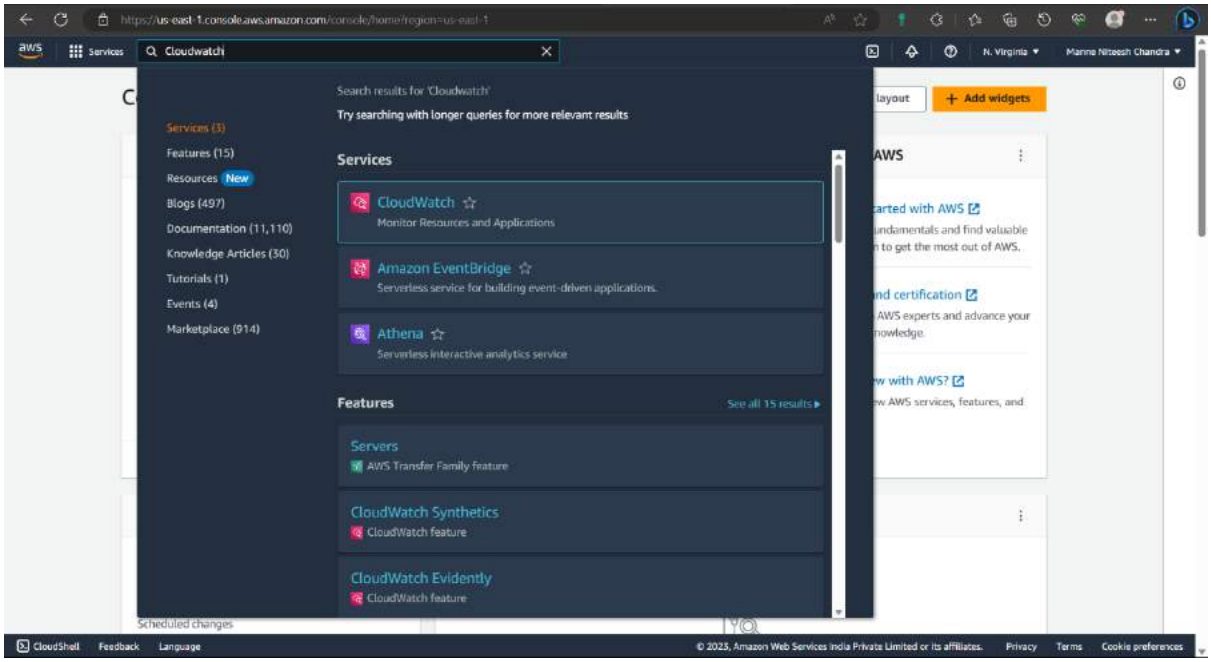


48)Now go to IAM user account and repeat the steps from 24-27. Now the bucket is created because we give administrator access in permissions which will let you create the tool use it, remove it.

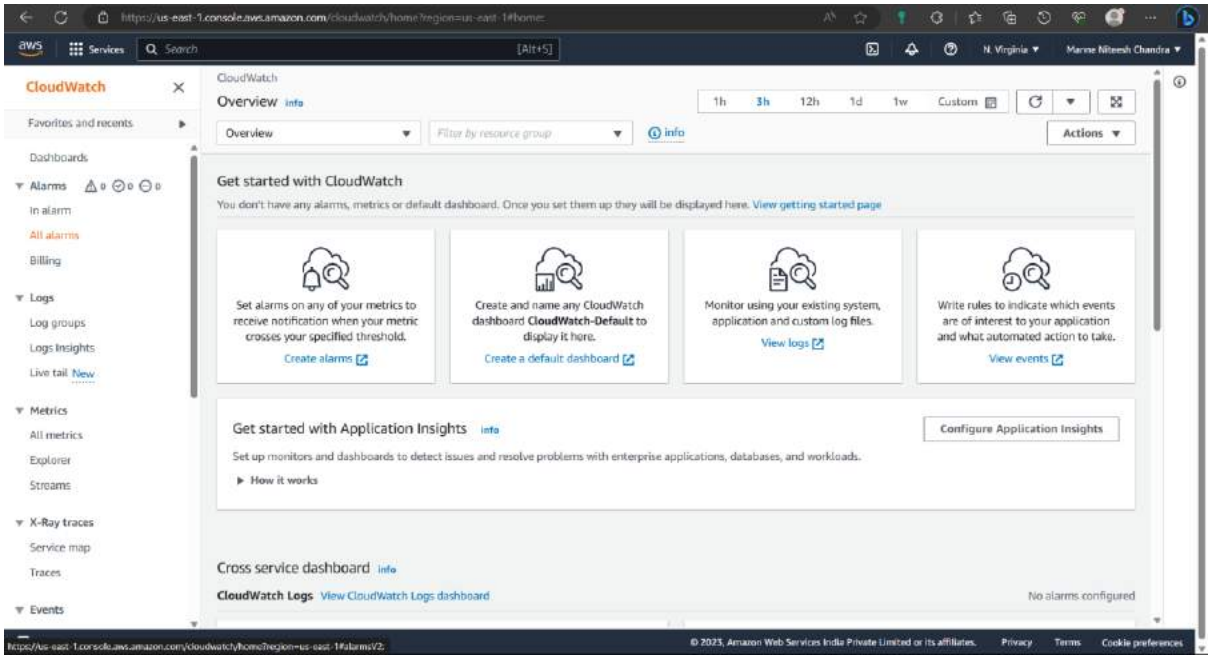


Billing Alarm:

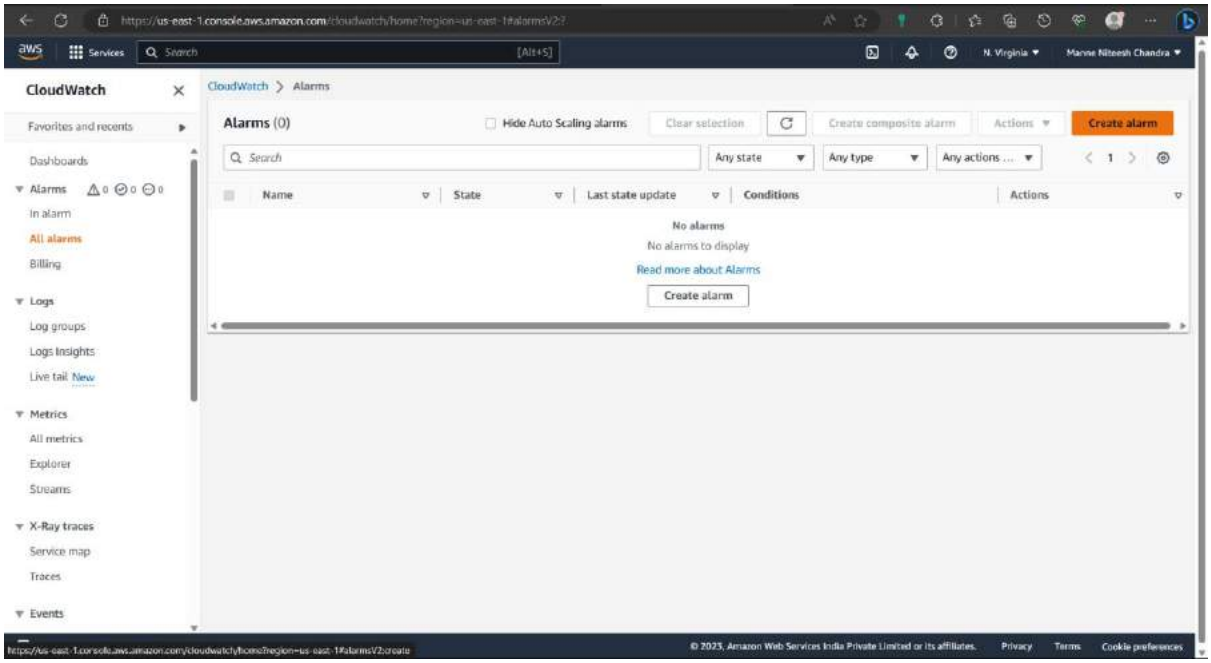
49)Type “CloudWatch” in search bar and select the “CloudWatch” service below.



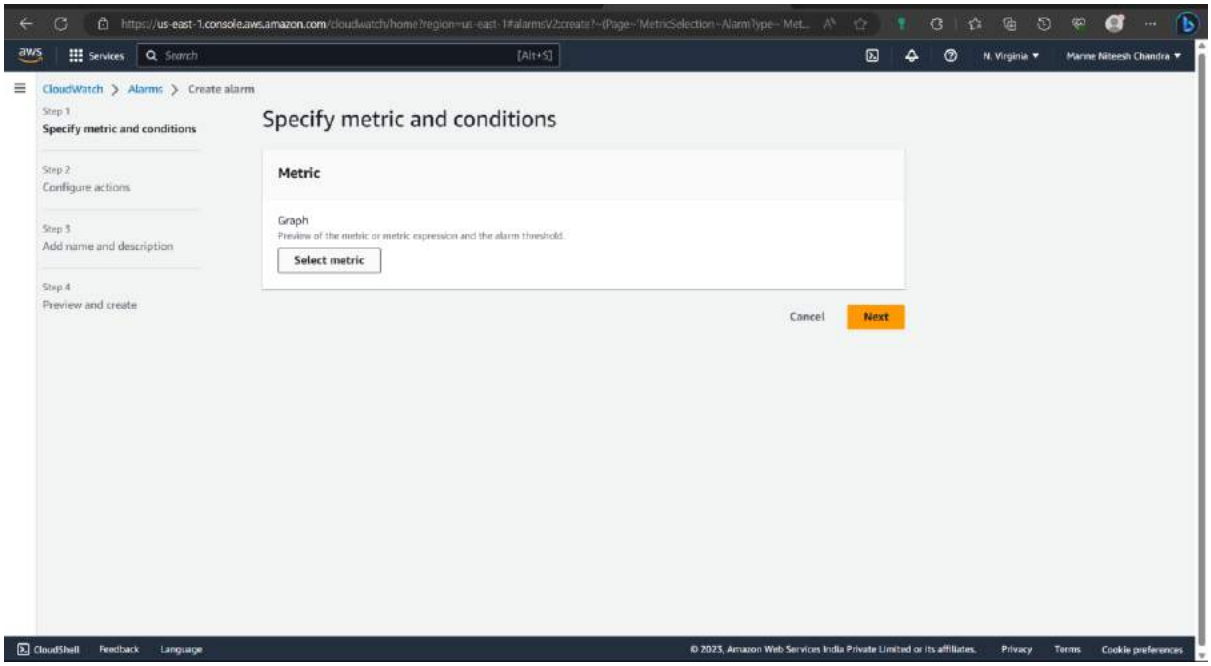
50)Select “All alarms” options in the navigation bar.



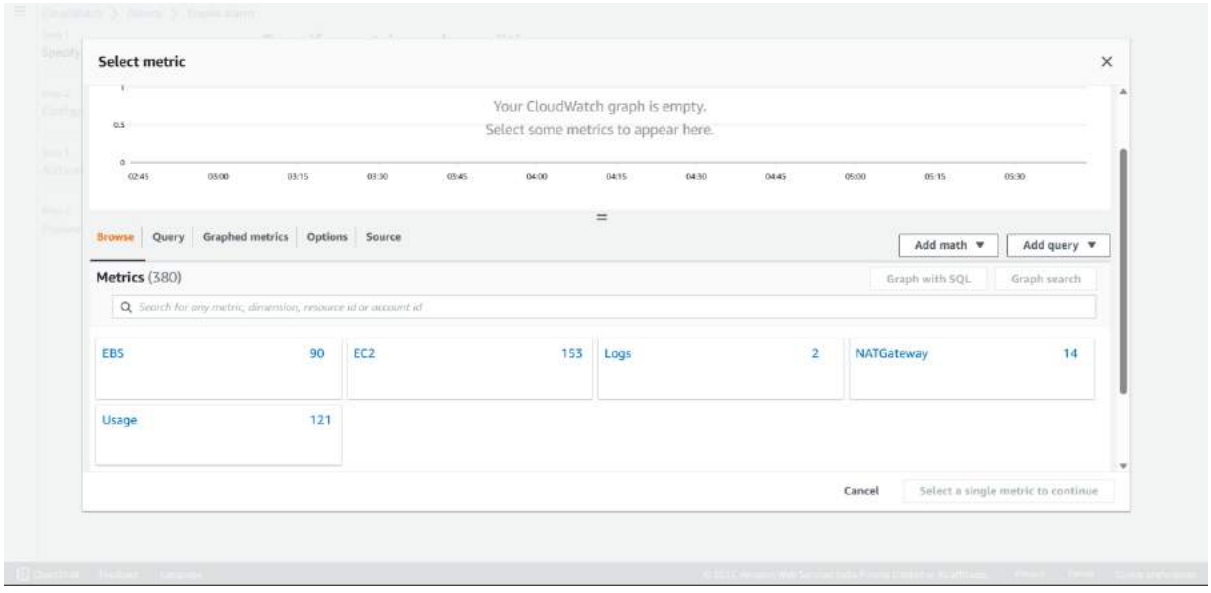
51)Click on “create alarm” option at top right side in yellow colour.



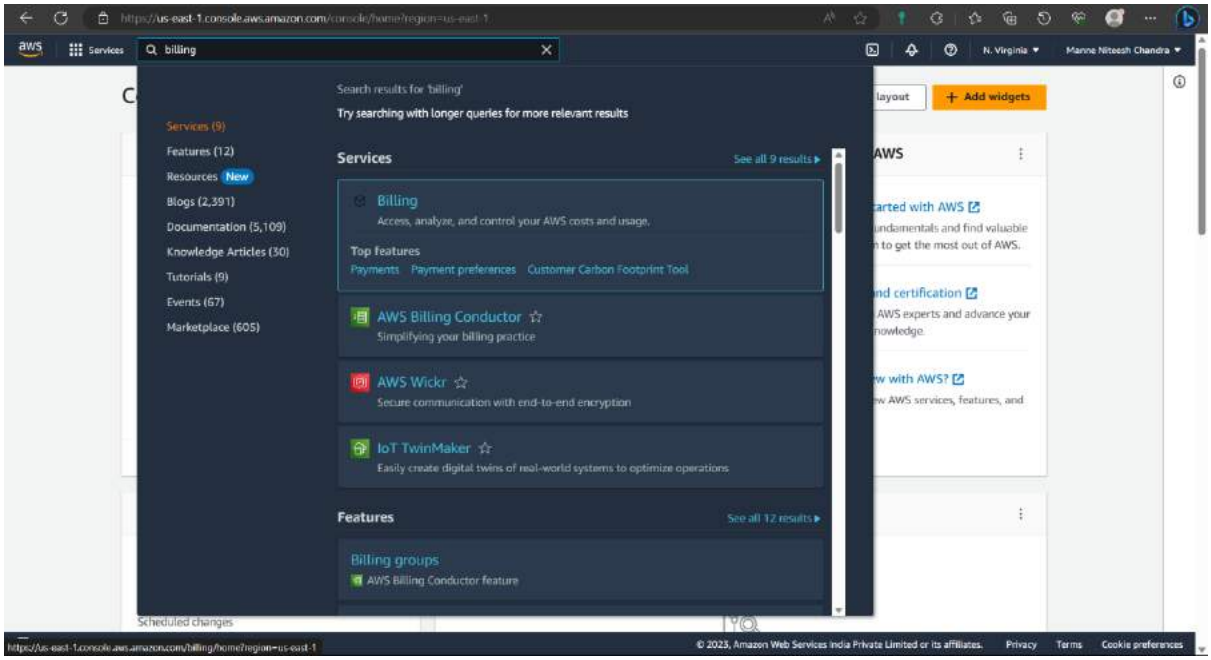
52)Click on “Select metric” option.



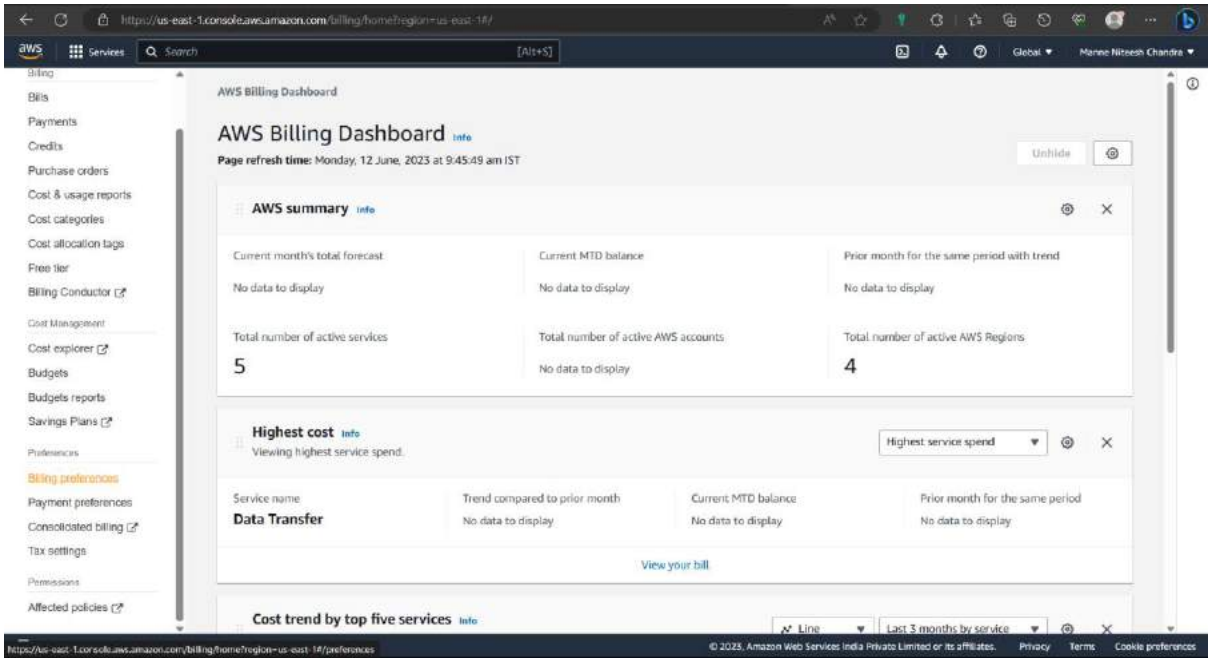
53)In “browse” we cannot find billing option. If you find the billing option click the billing option, else follow the below steps to enable the billing option.



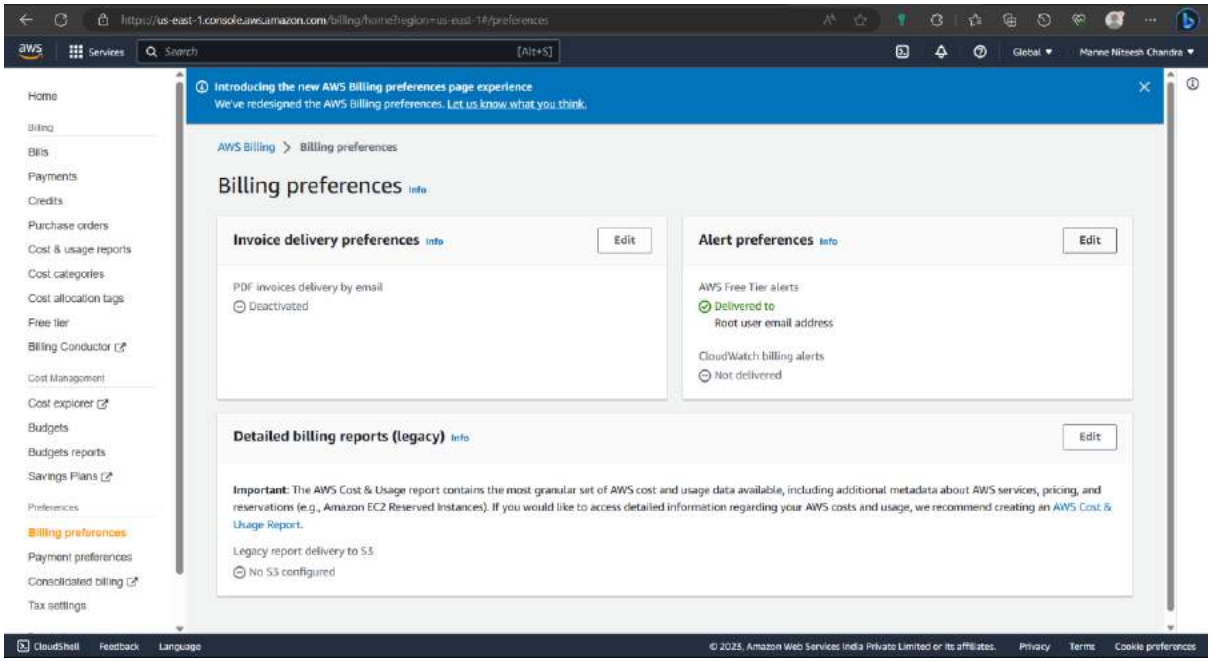
54)Now go to search bar type “Billing” and select the “Billing” option.



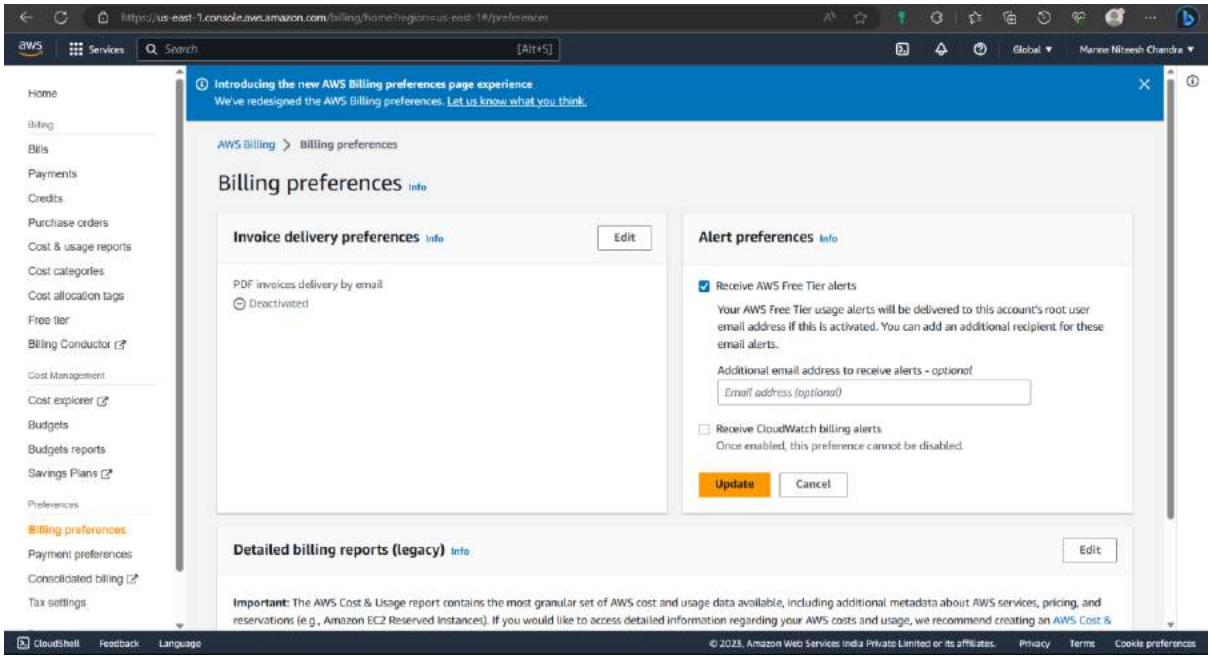
55)Select the “Billing Preference” in navigation panel i.e., at bottom of the navigation panel.



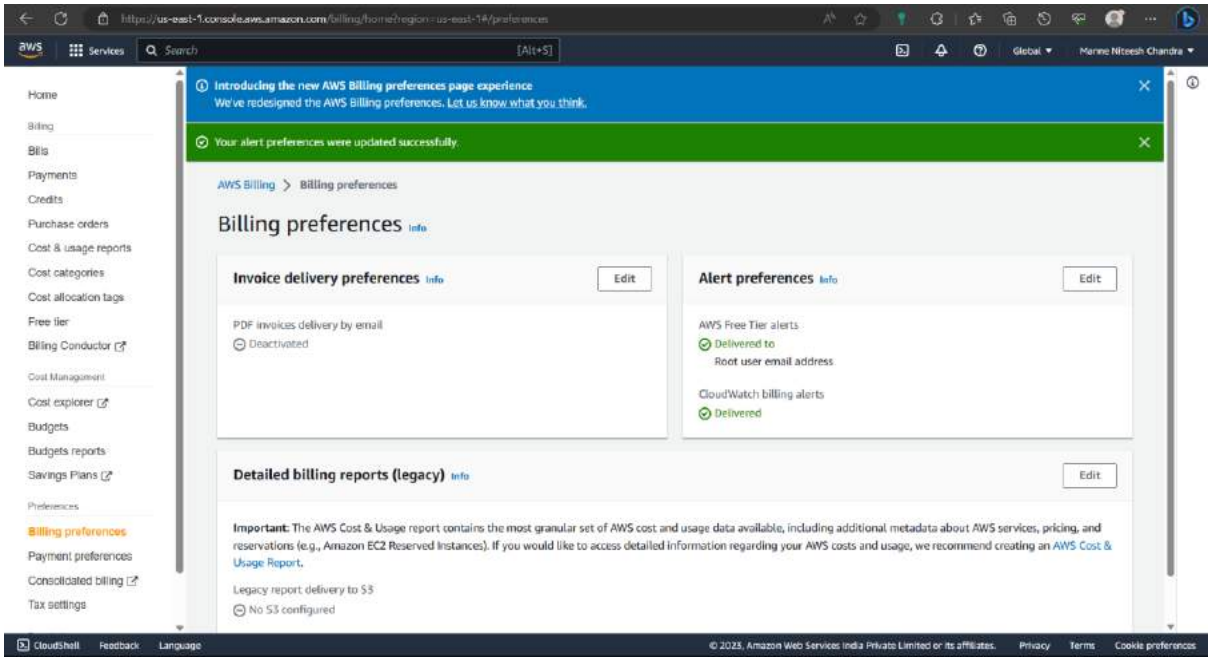
56)Click on edit option for “Alert preferences”.



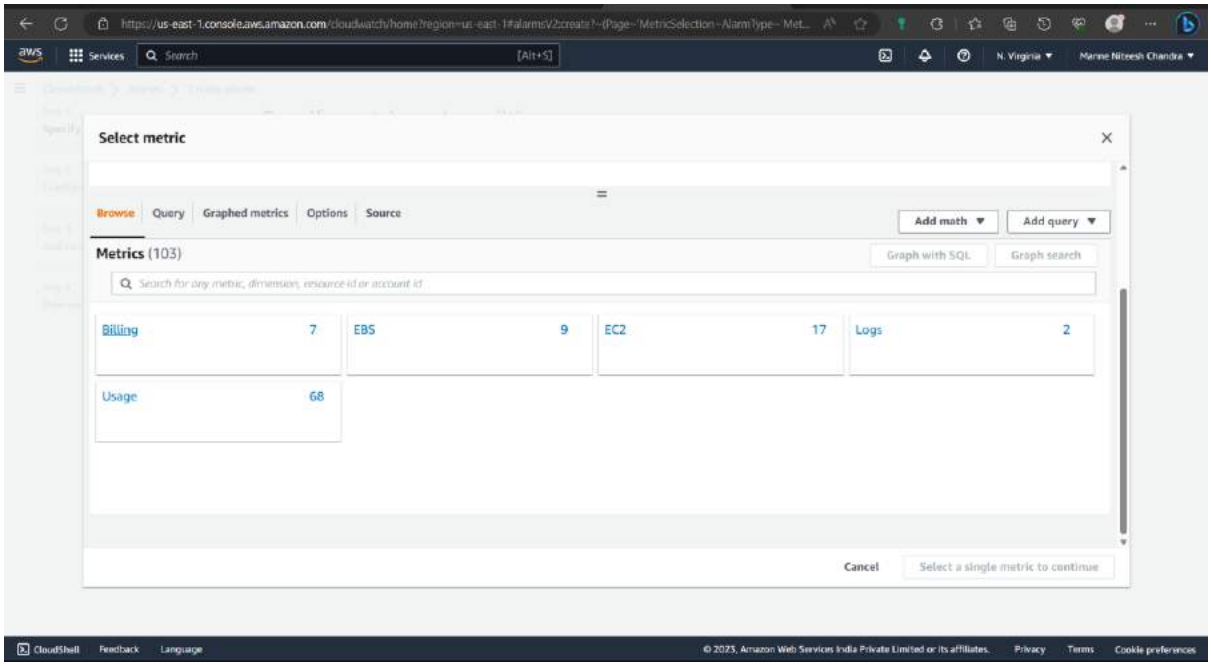
57)Enable the “CloudWatch billing alerts” option (Note: Once it is enables it cannot be disabled) click update.



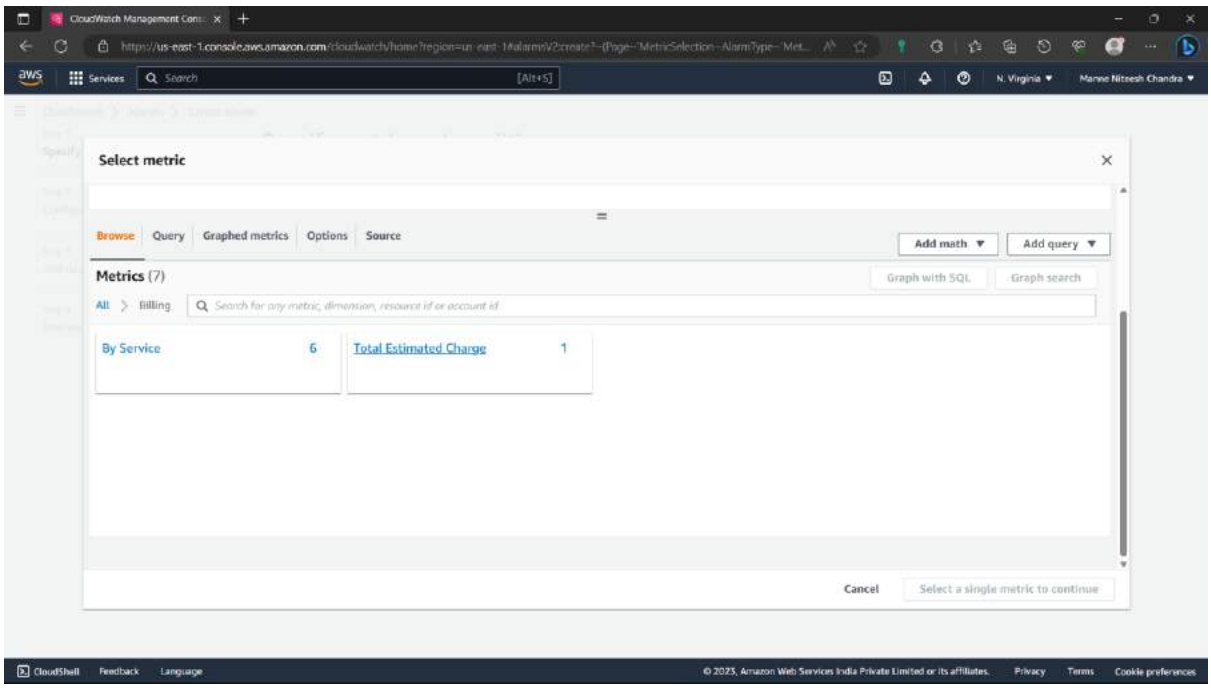
58)Now it is enabled.



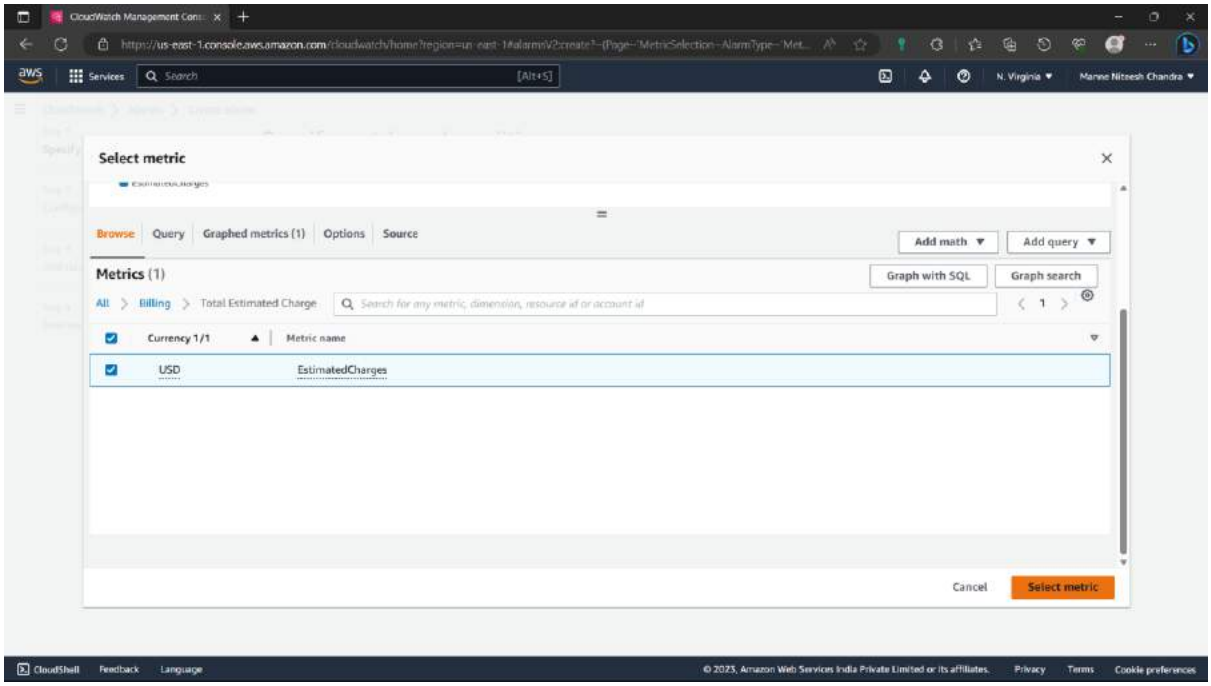
59)Now repeat the steps 49-52 and in browse option we can see the billing option, Click the “Billing” option.



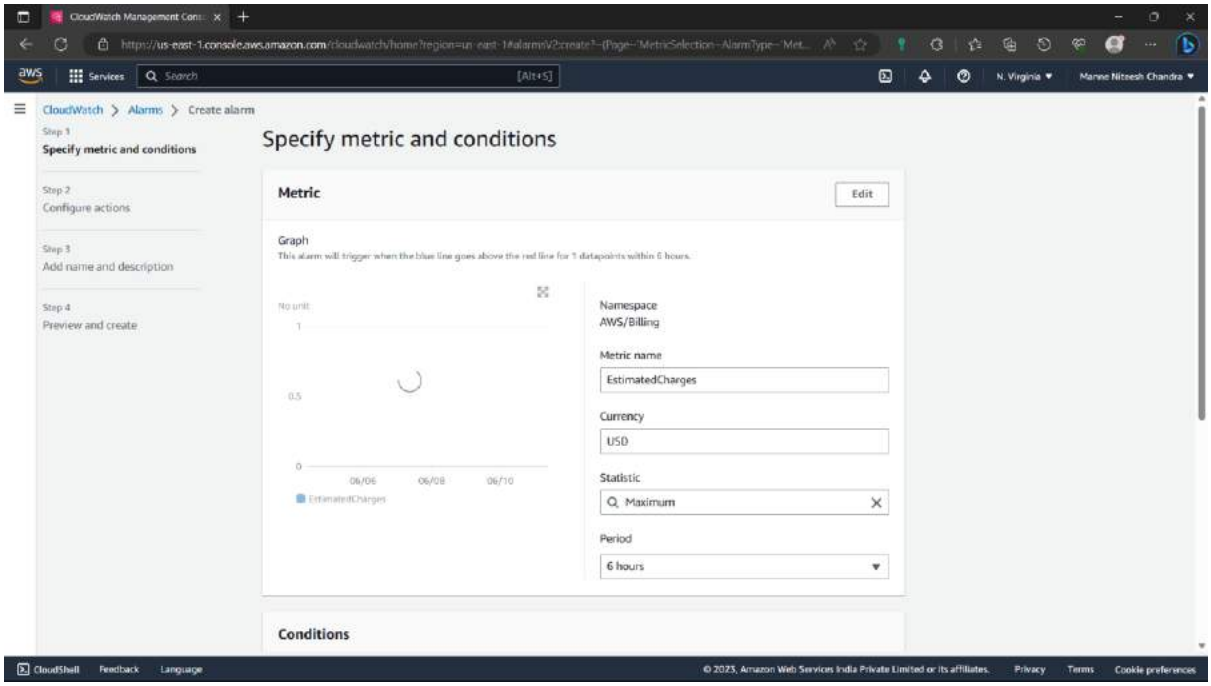
60)Click the “Total Estimated Charge”.



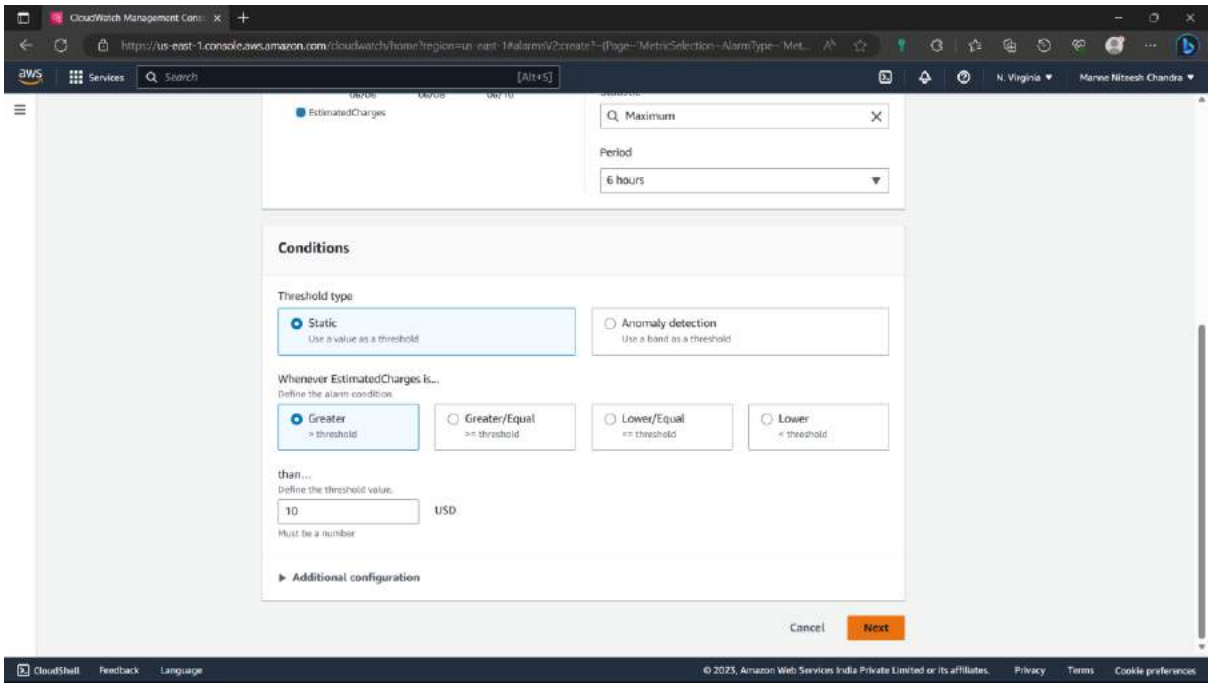
61)Select the “EstimatedCharges” and click on select metric.



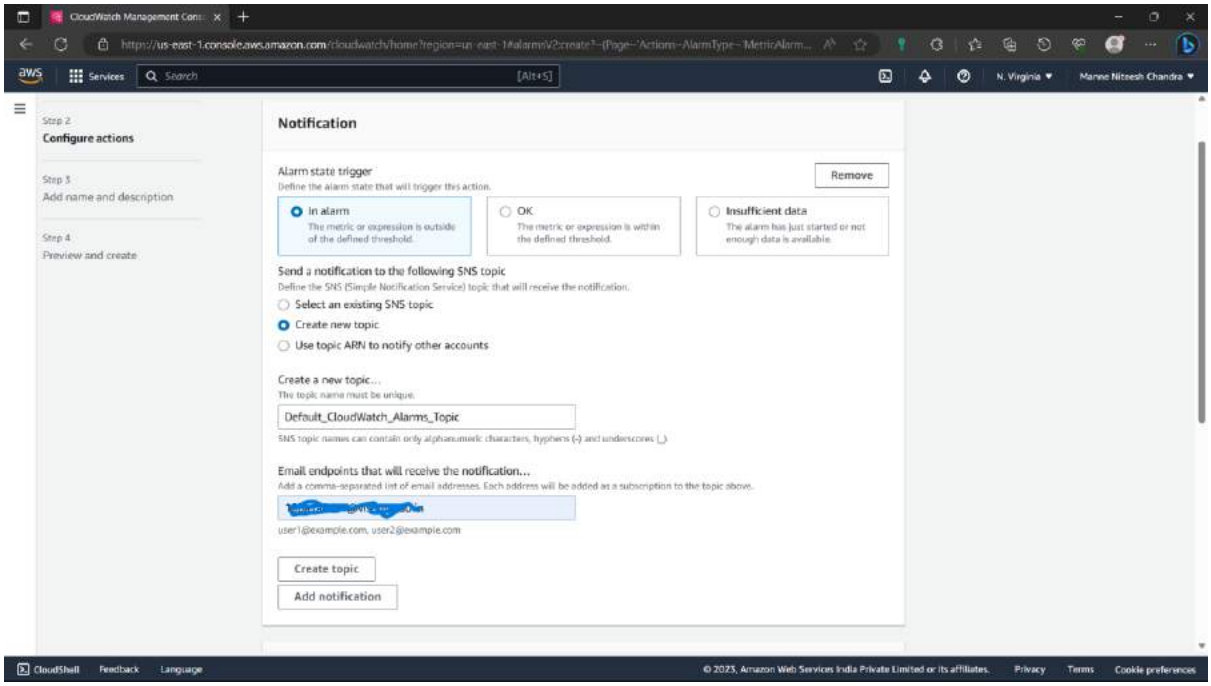
62)Now we can see the graph give an name to metric, Static-maximum, period.



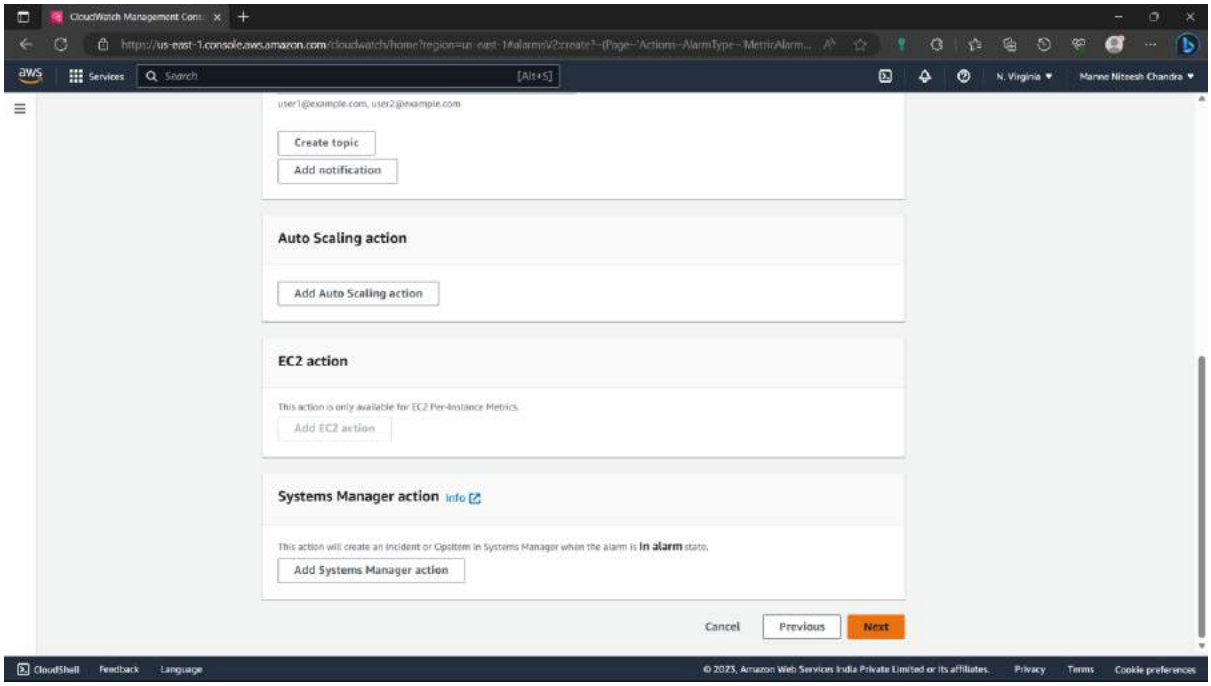
63)In conditions Threshold type-Static, EstimatedCharges-Greater, than-10(value your choice i.e., bill should not exceed this value) and click on “Next”.



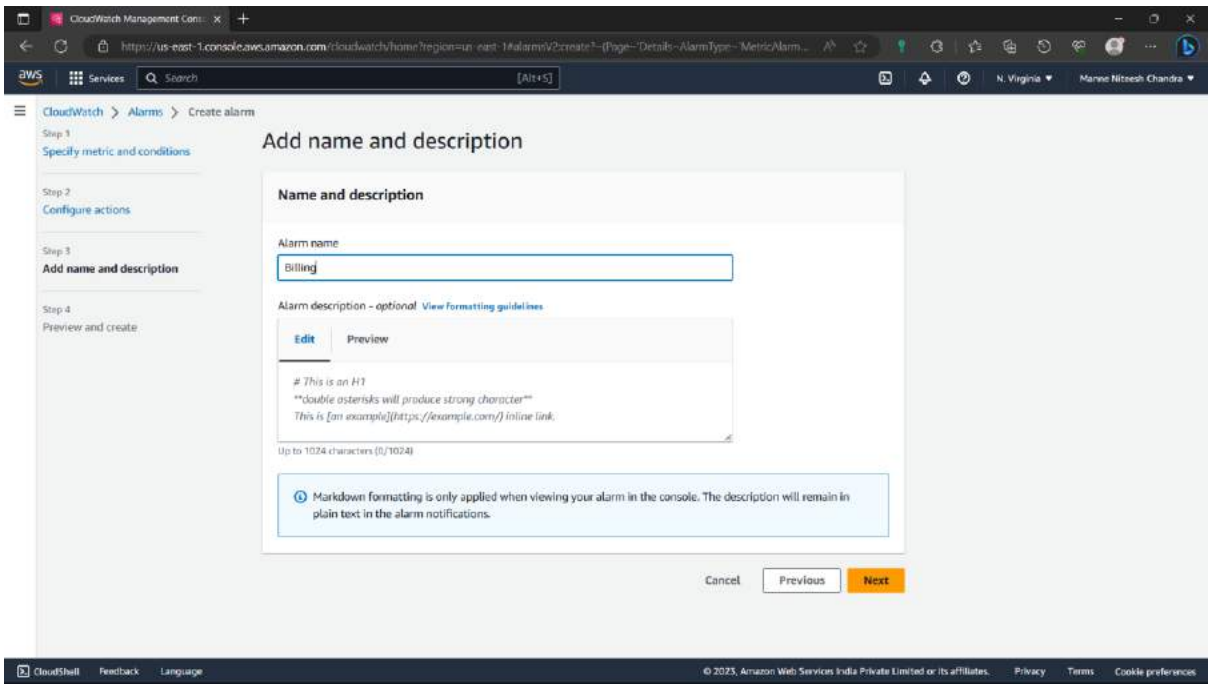
64)Notification- In alarm, Notification to following SNS topic- Create new topic, Name, Email ID(which email id you want receive the notification about billing).



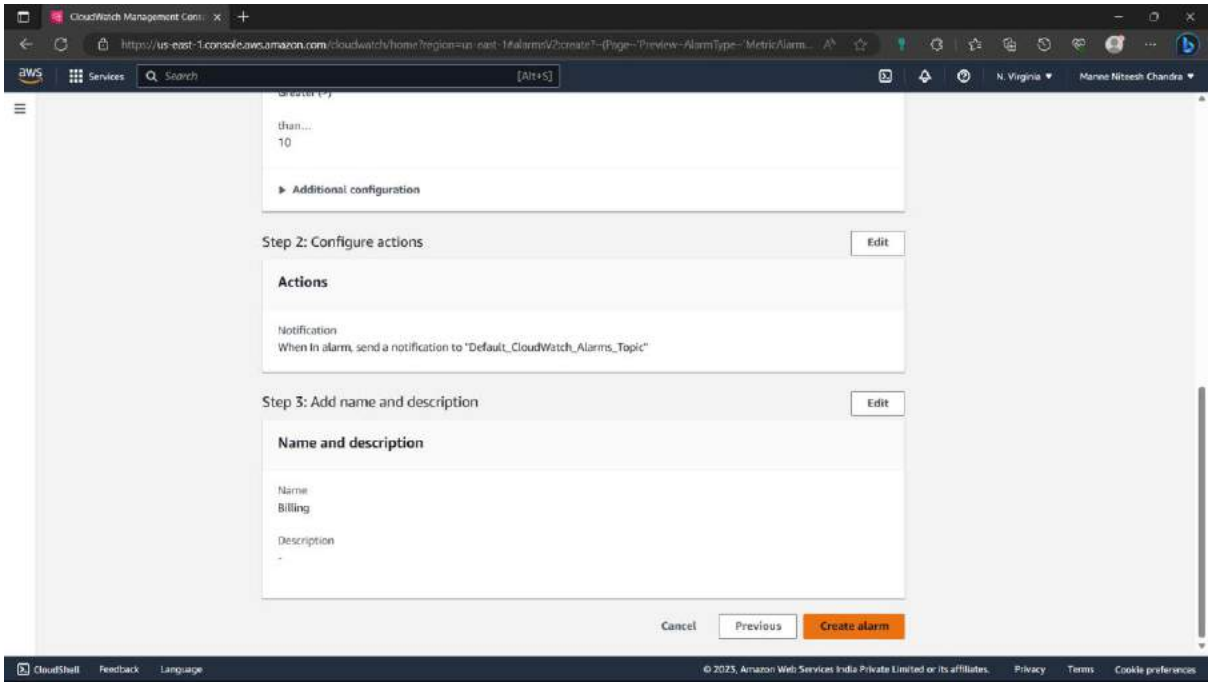
65)Click on “Next”.



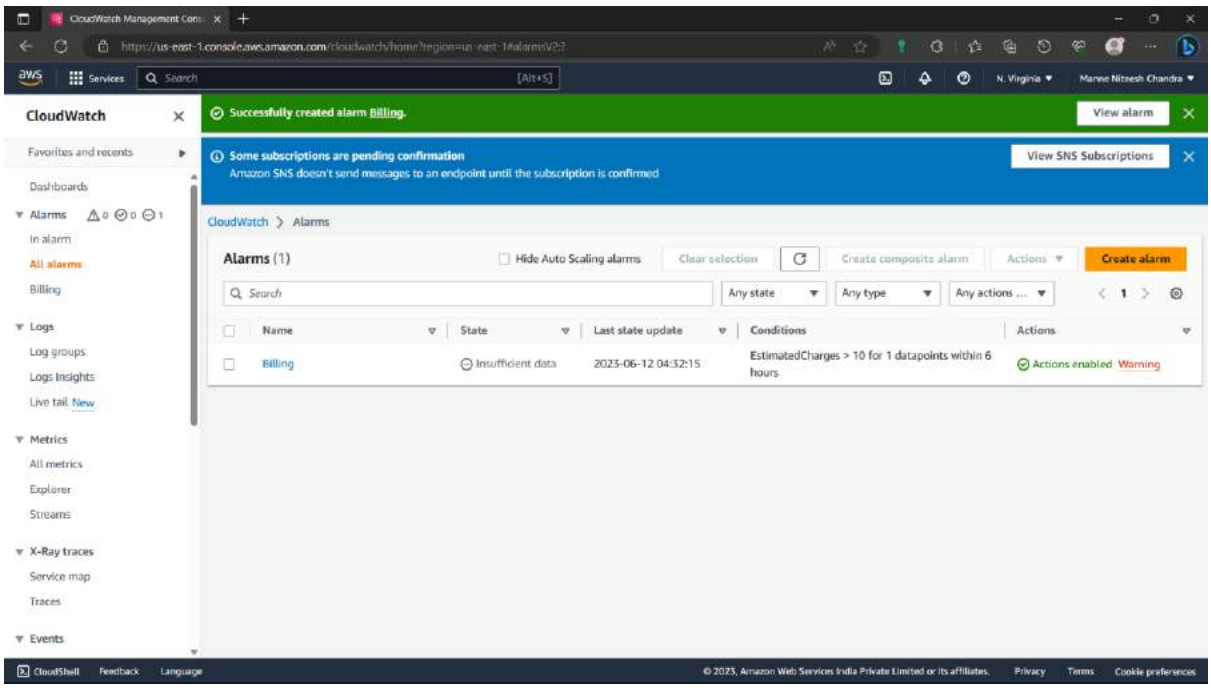
66)Give a name, description of your choice and click on next.



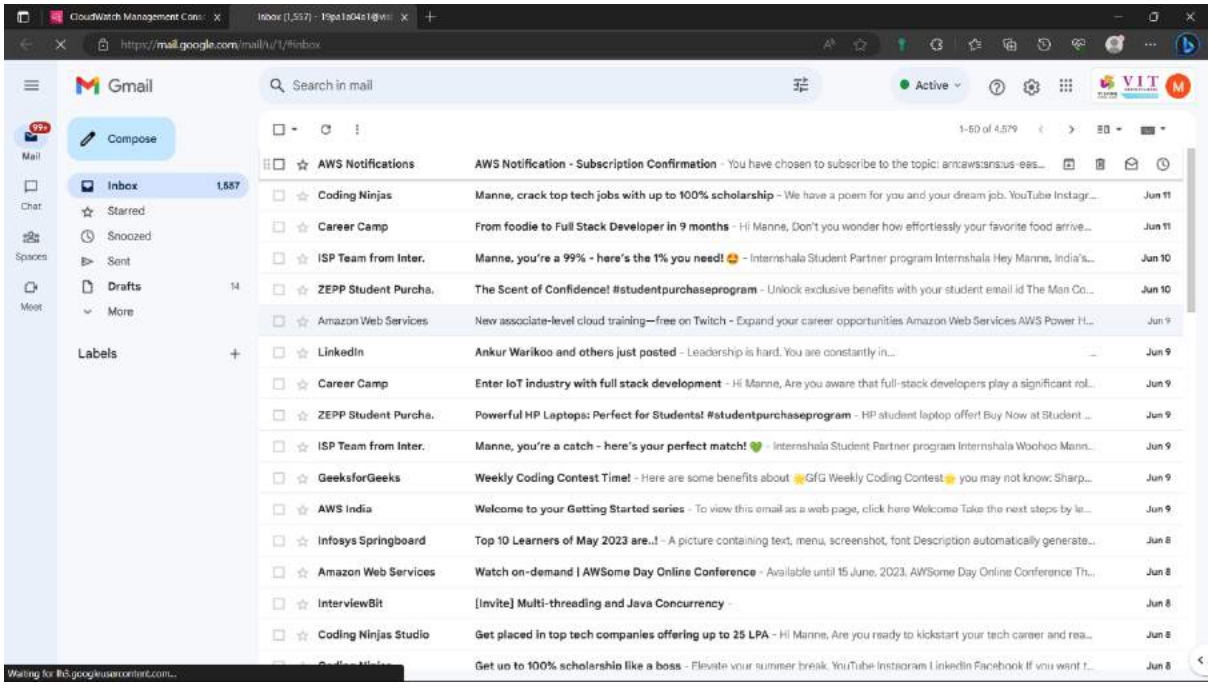
67)Review all the data and click on “Create alarm” option.



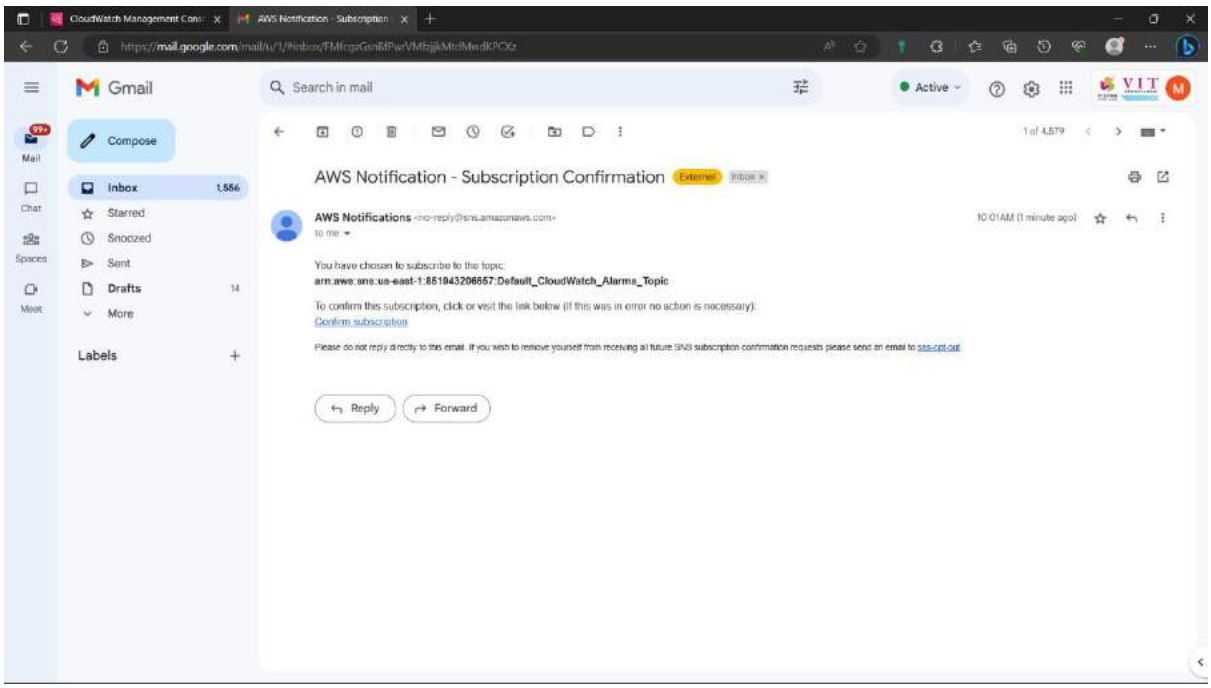
68)Now the billing alarm is created.



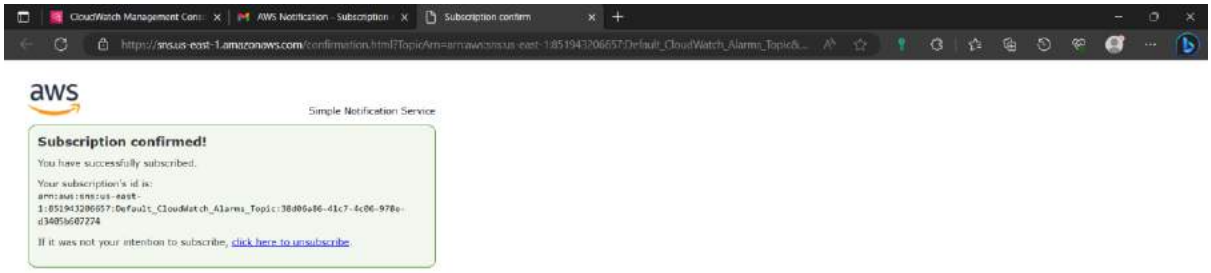
69)Open the Gmail of the email id you provided in the billing for notification. you can see there is AWS notification open it.



70)Click the “confirm subscription” option.

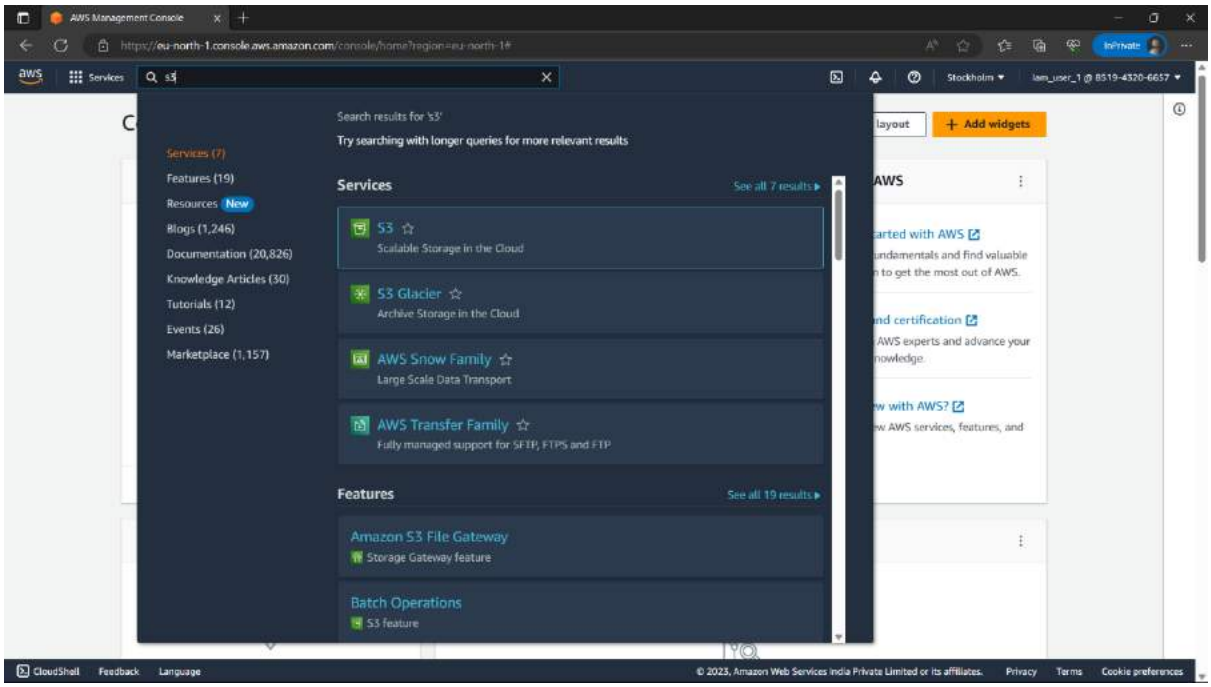


71)Now you will receive the notification if the bill value exceeds the given value.

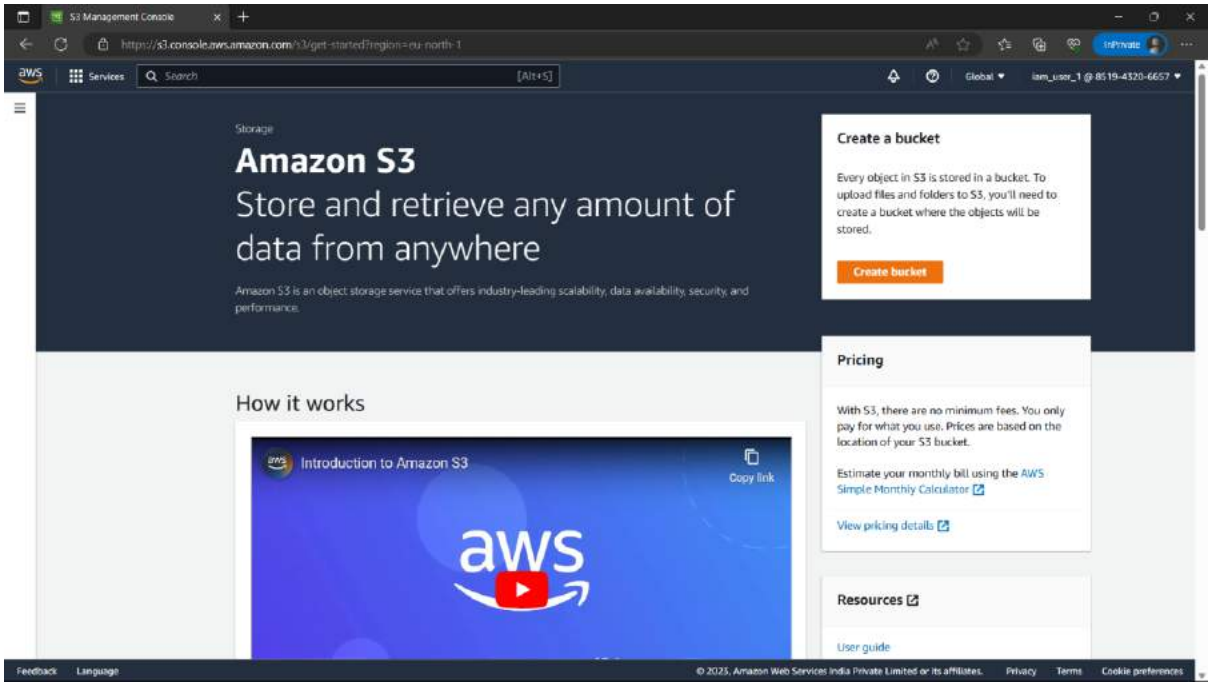


S3 Bucket:

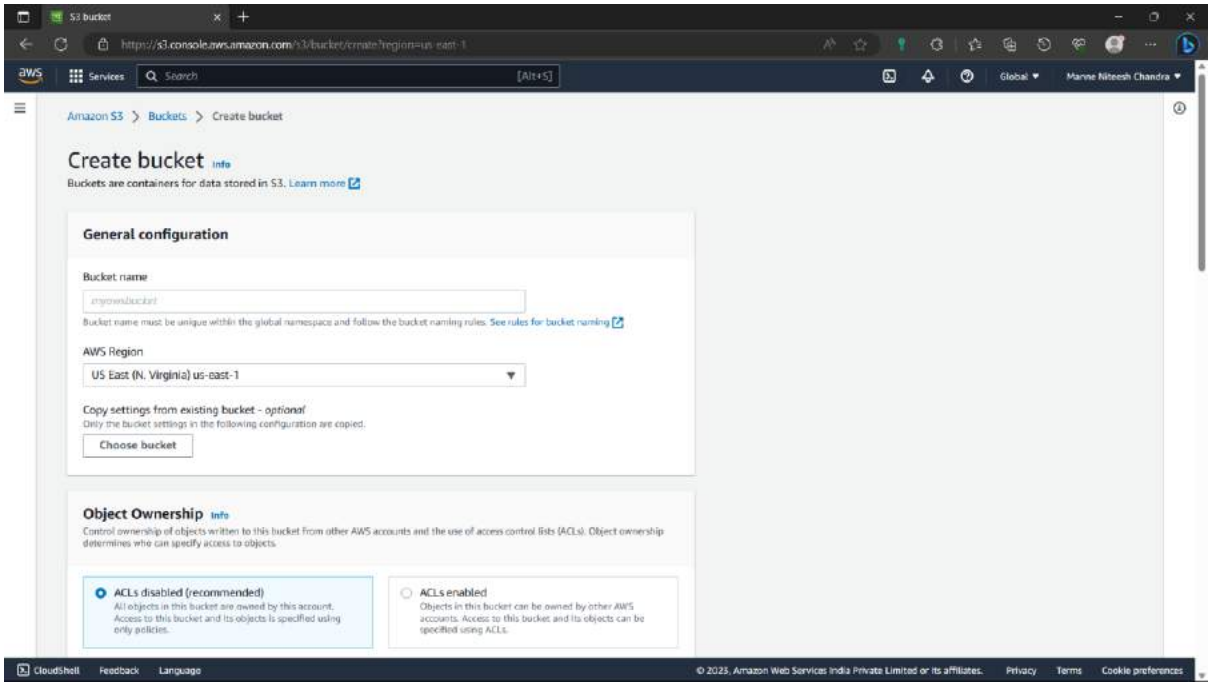
72)Type “S3” in search bar and select the “S3” option below.



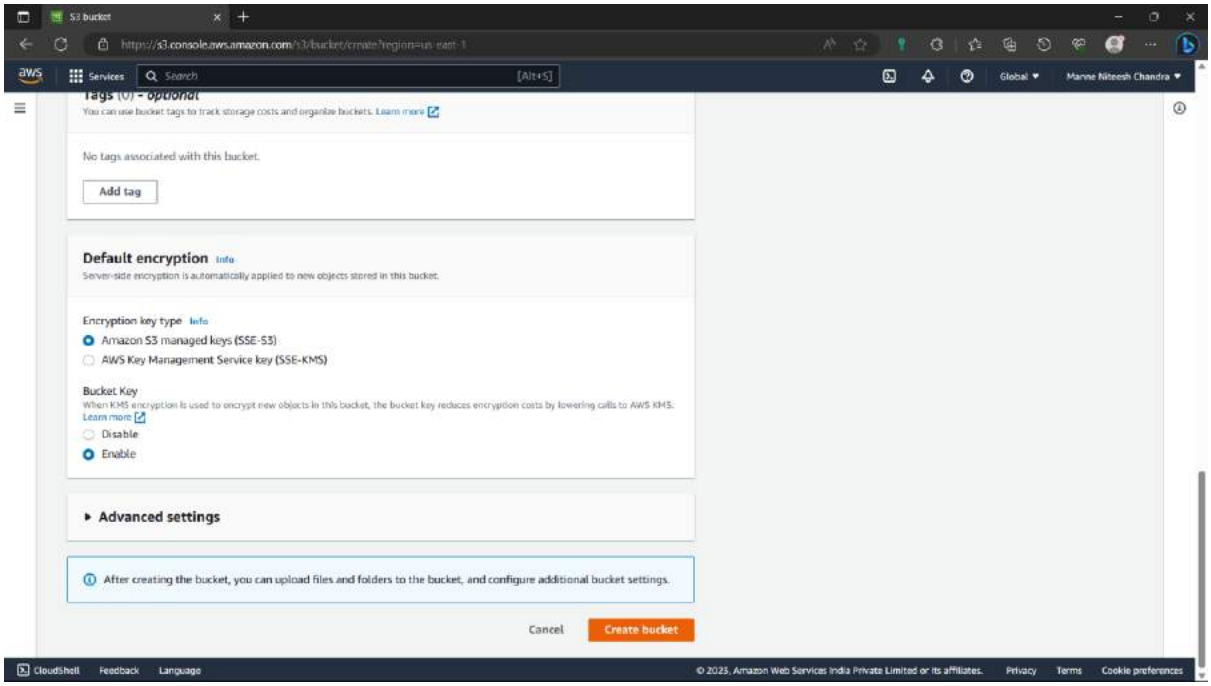
73)Click “create bucket” option.



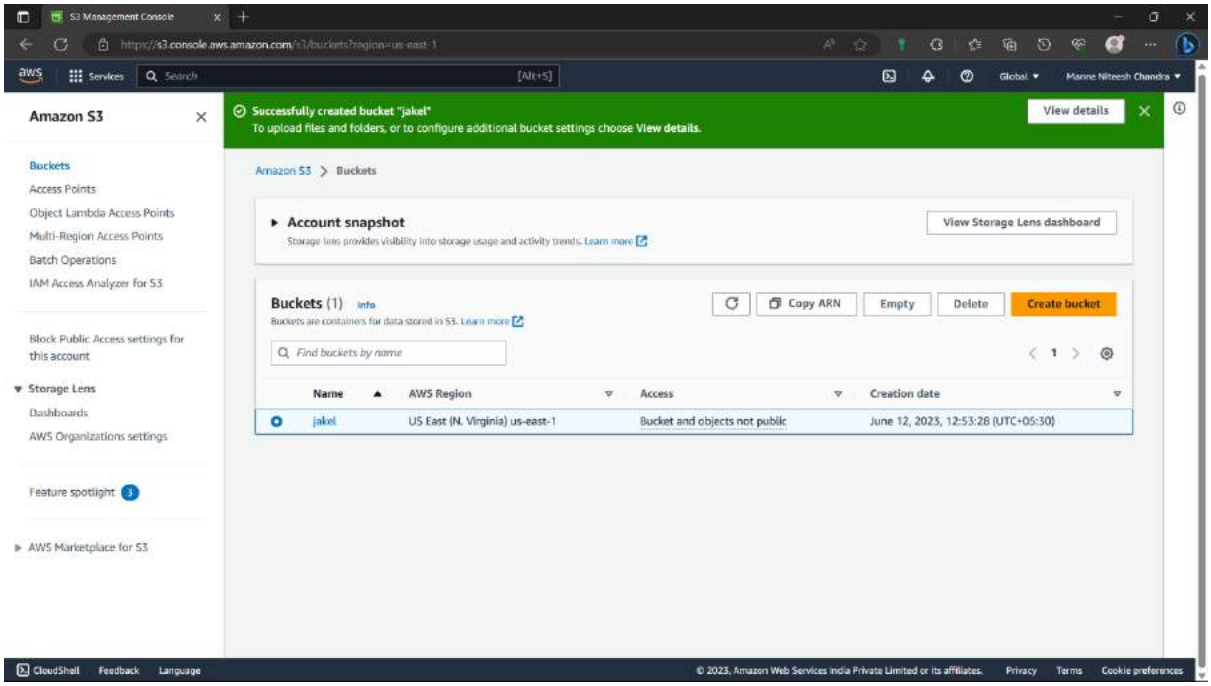
74)Give a unique name to the bucket.



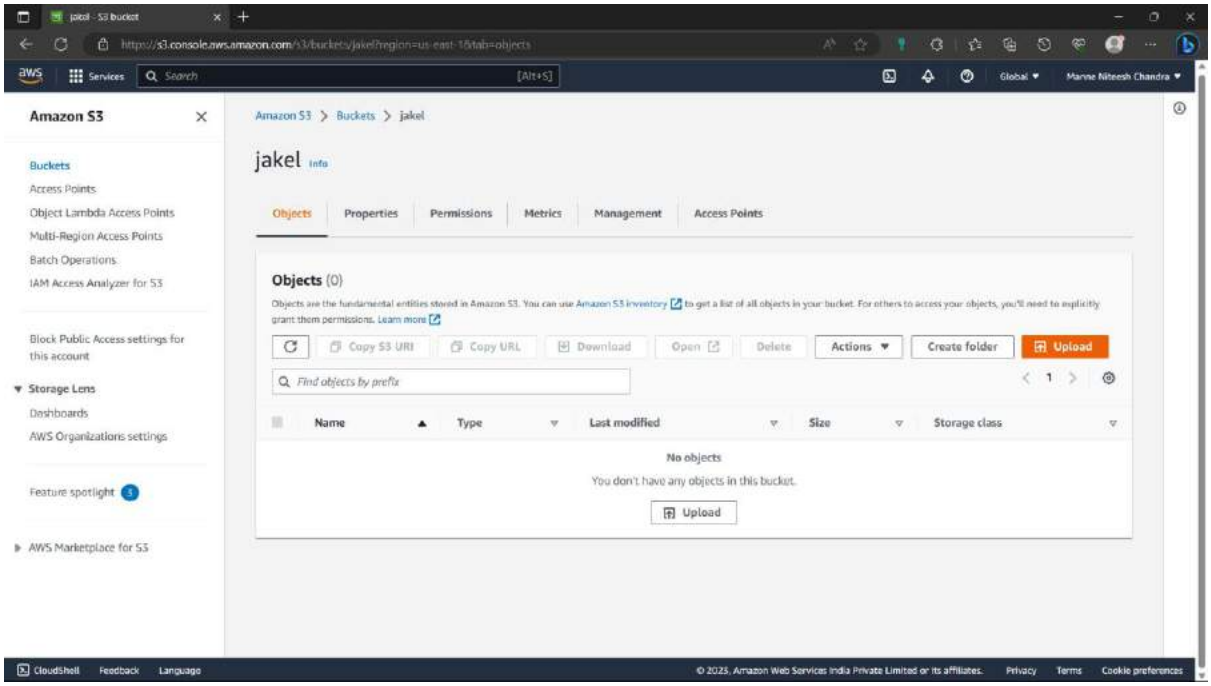
75)Click on “create bucket”



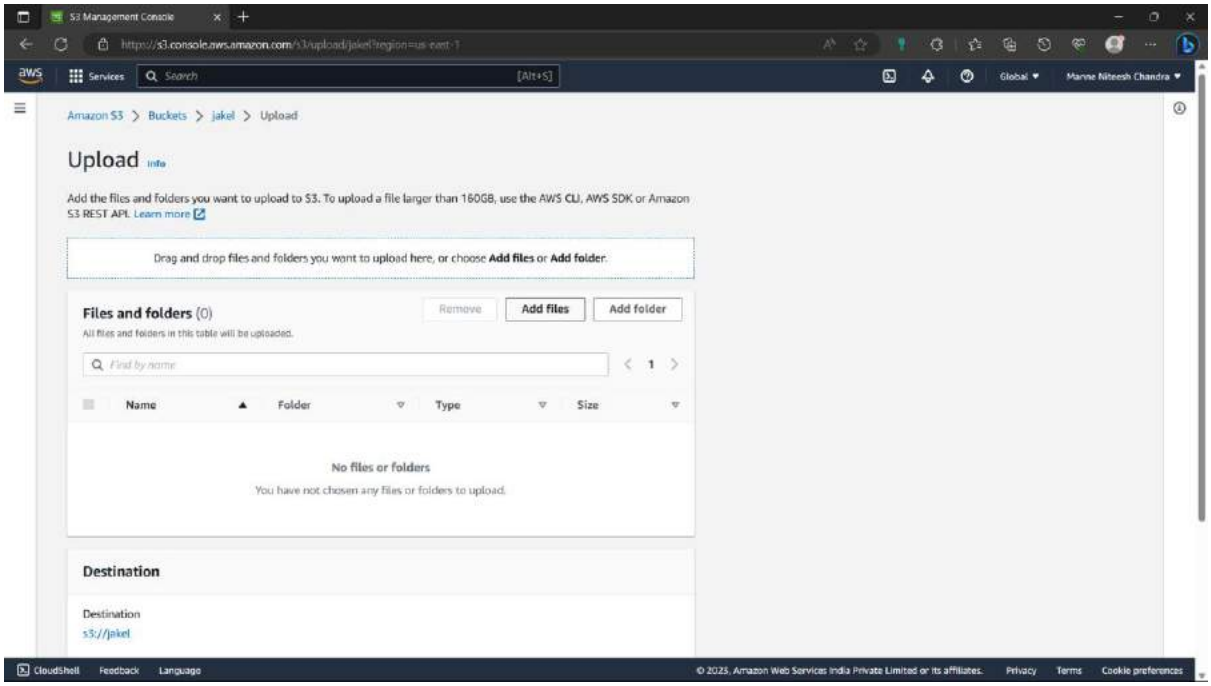
76)Now a “S3” bucket is created.



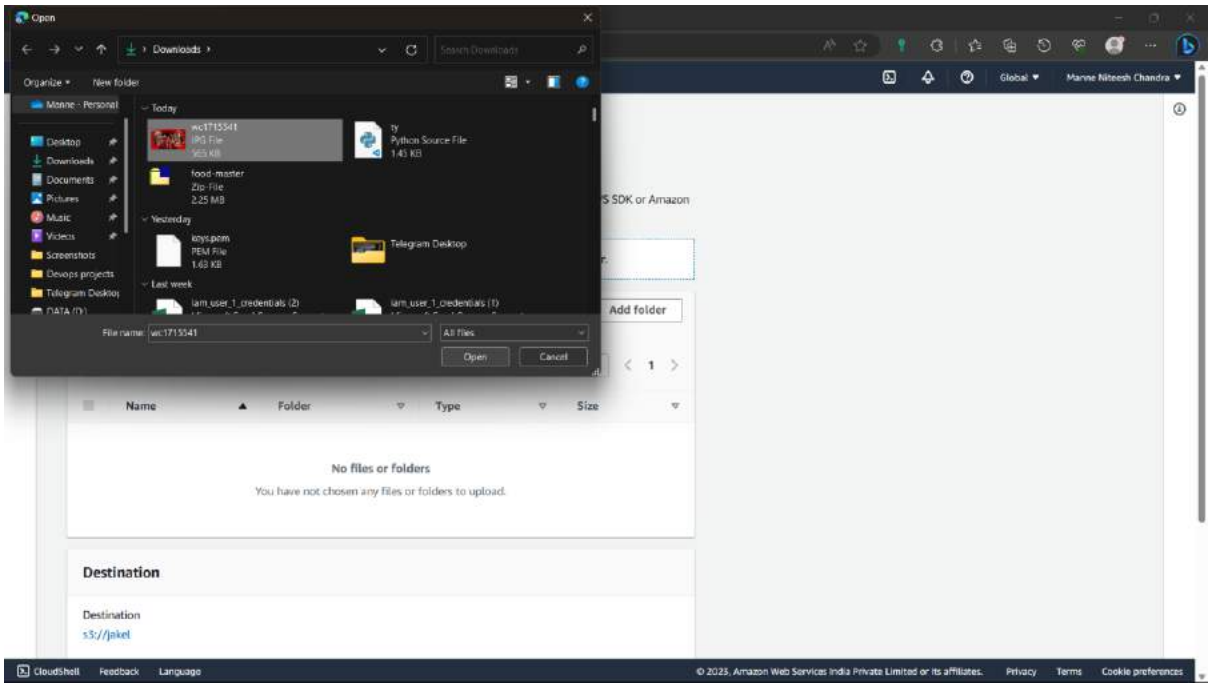
77)Now open the S3 bucket you created and click on “upload”.



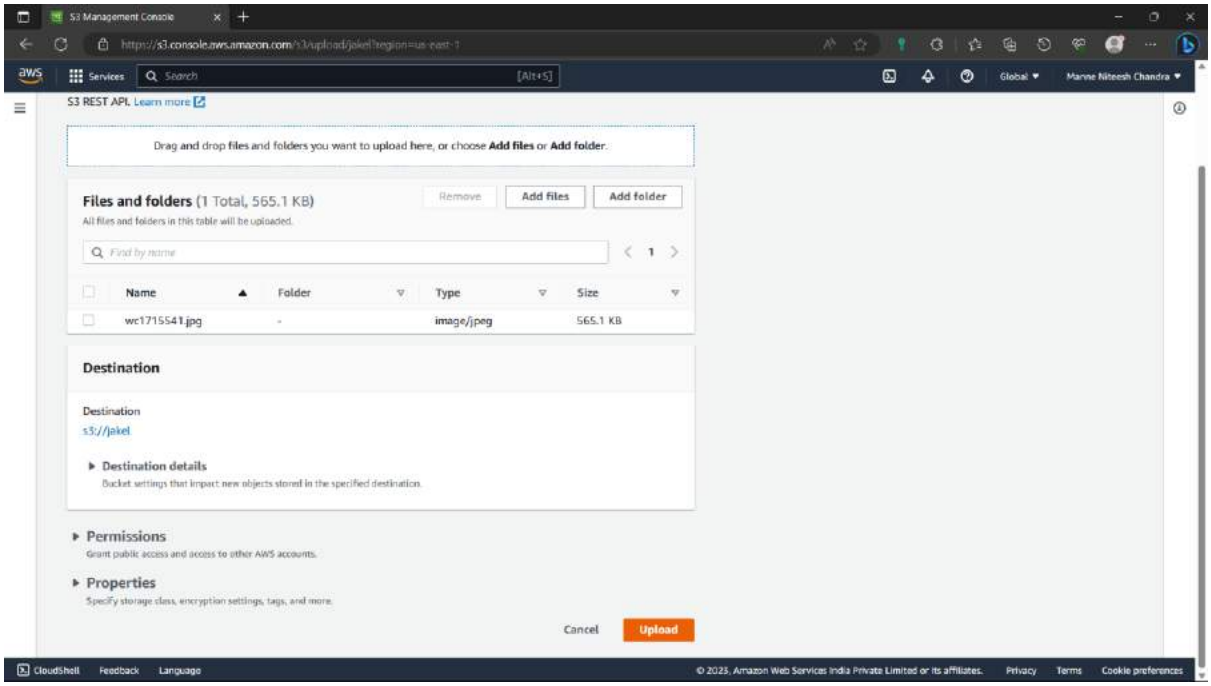
78)Click on “Add file” option.



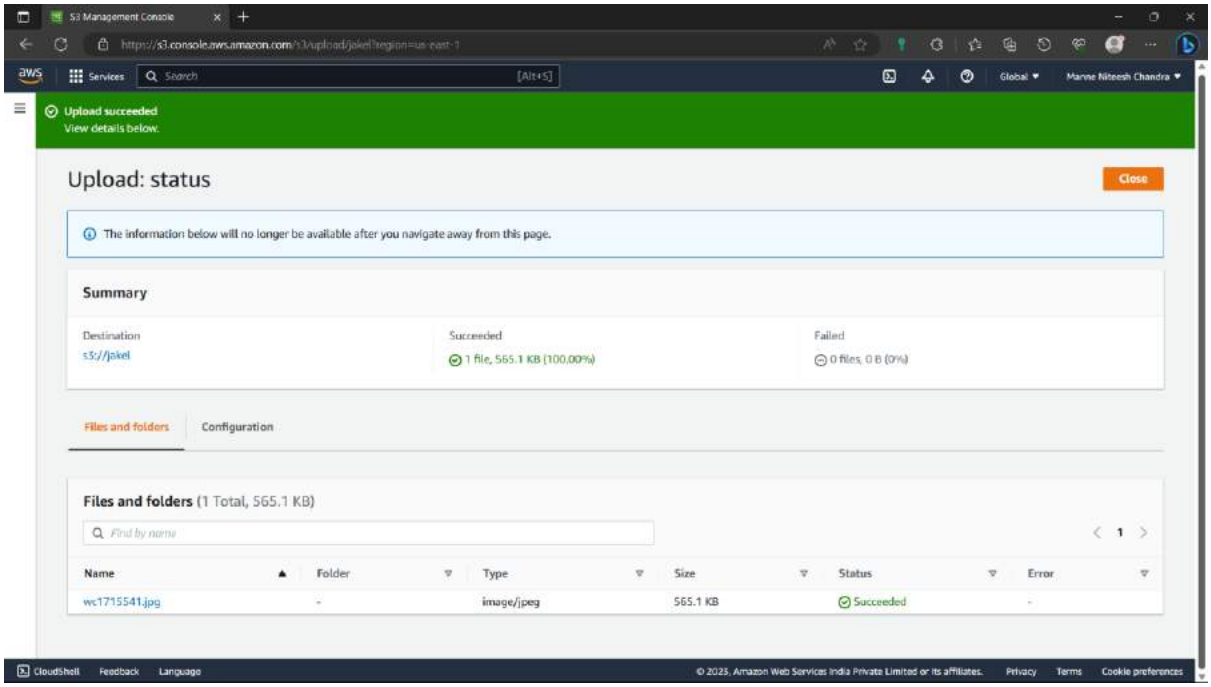
79)A new window of local computer open to choose which file to be upload, click on the file or folder you want to upload.



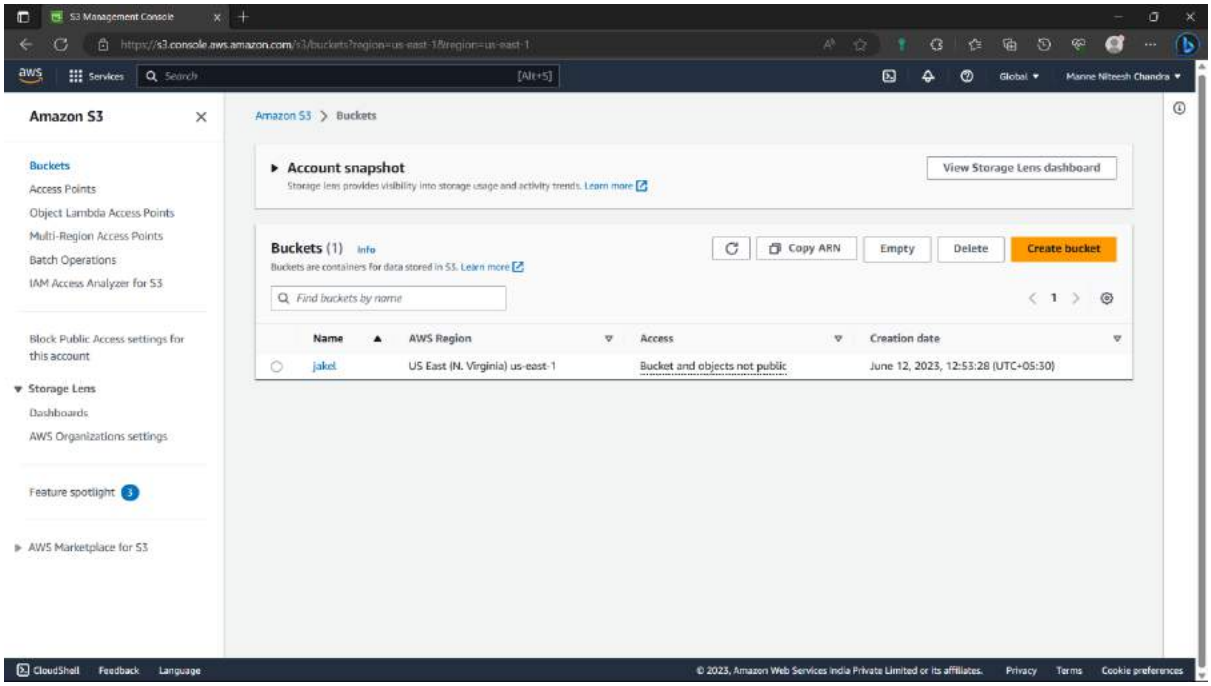
80)Click on upload.



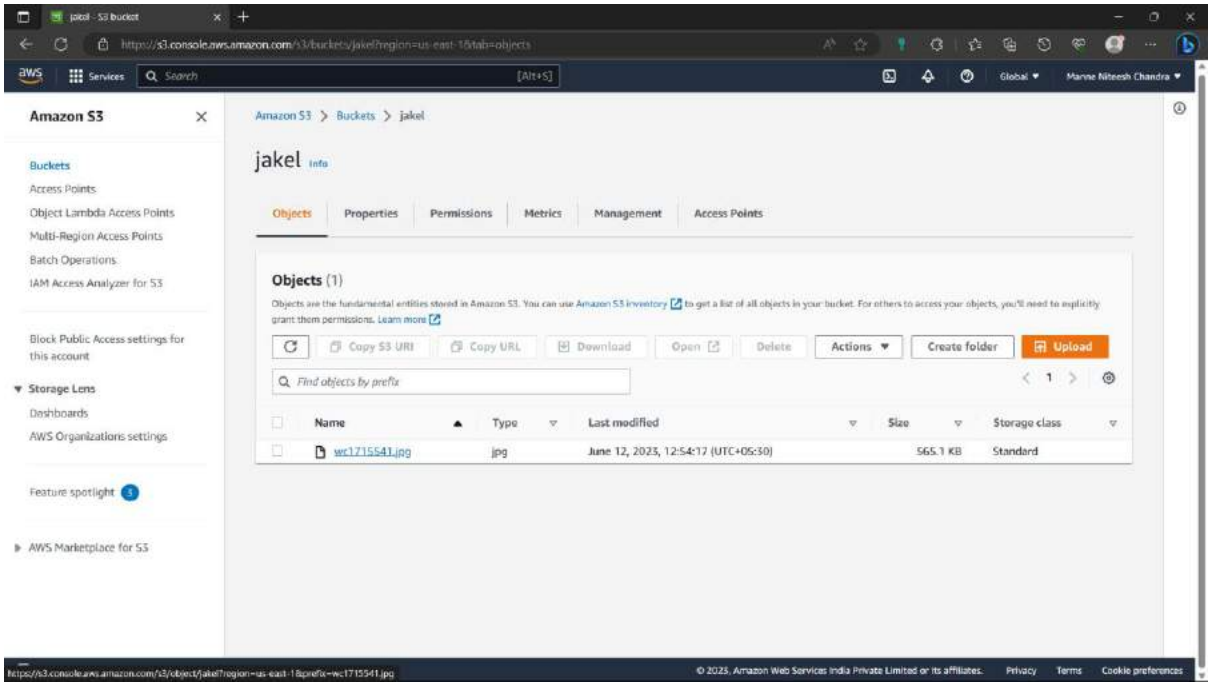
81)The file has been uploaded. Click on close option at the top right.



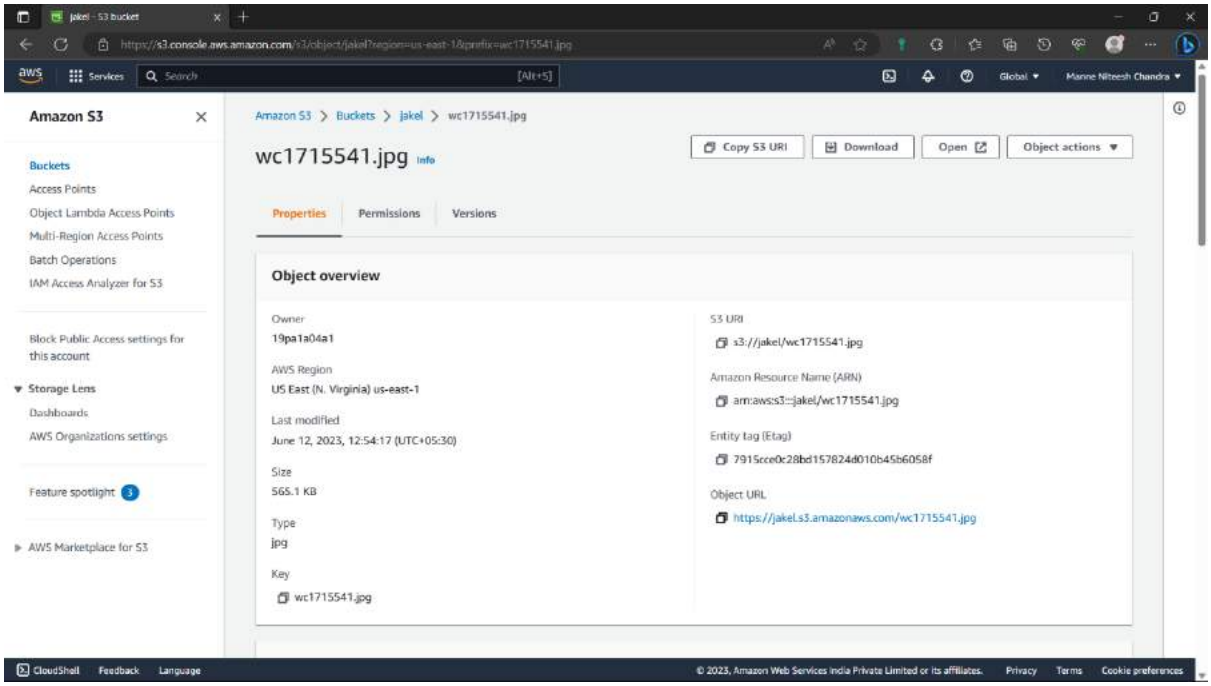
82)Now you returned to the buckets page.



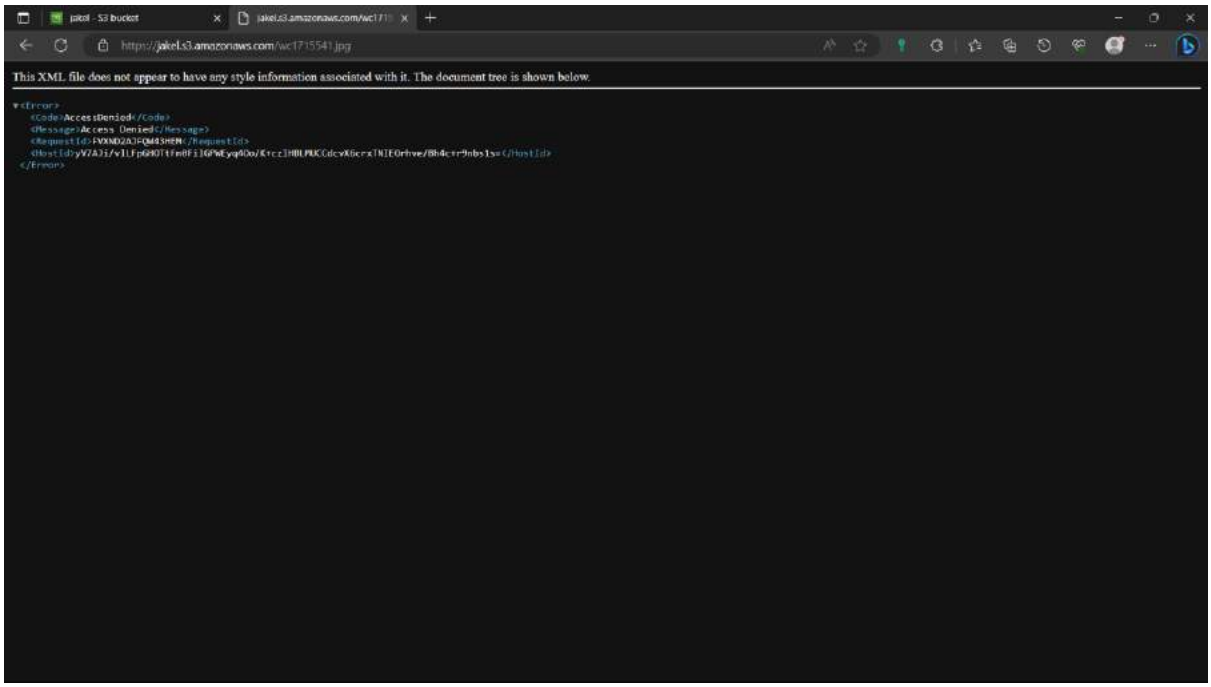
83)Open the bucket you created, click on the file you uploaded.



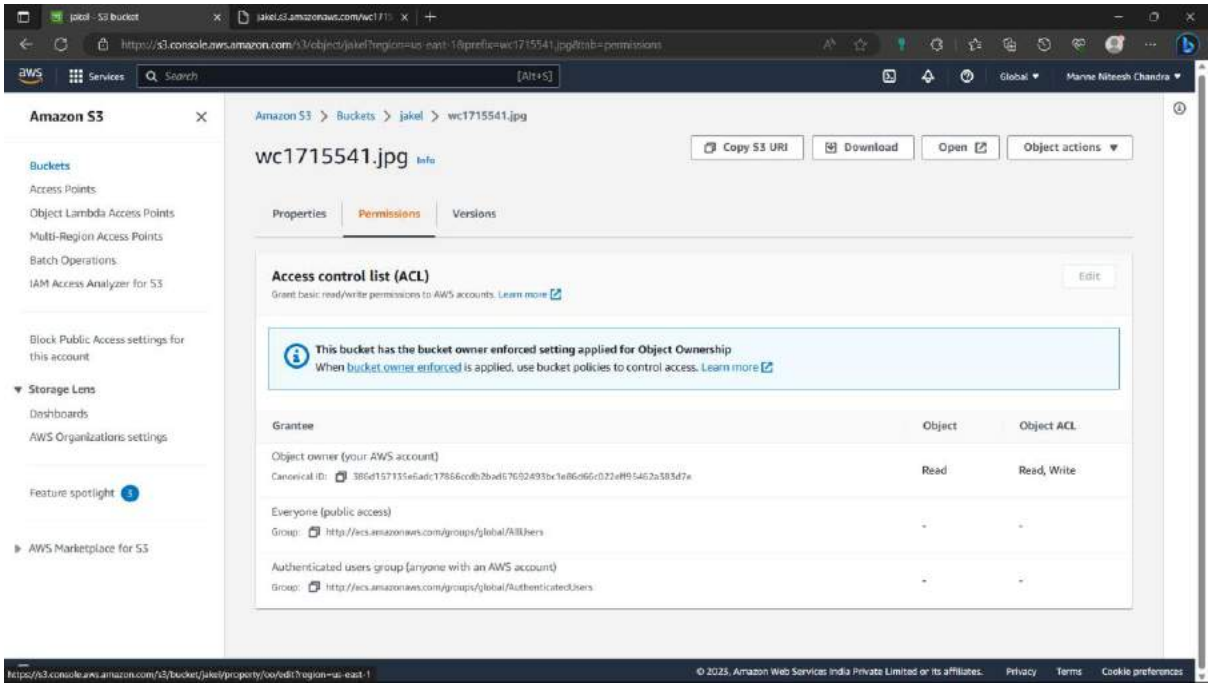
84)Now copy the “Object URL” and paste it in a new window.



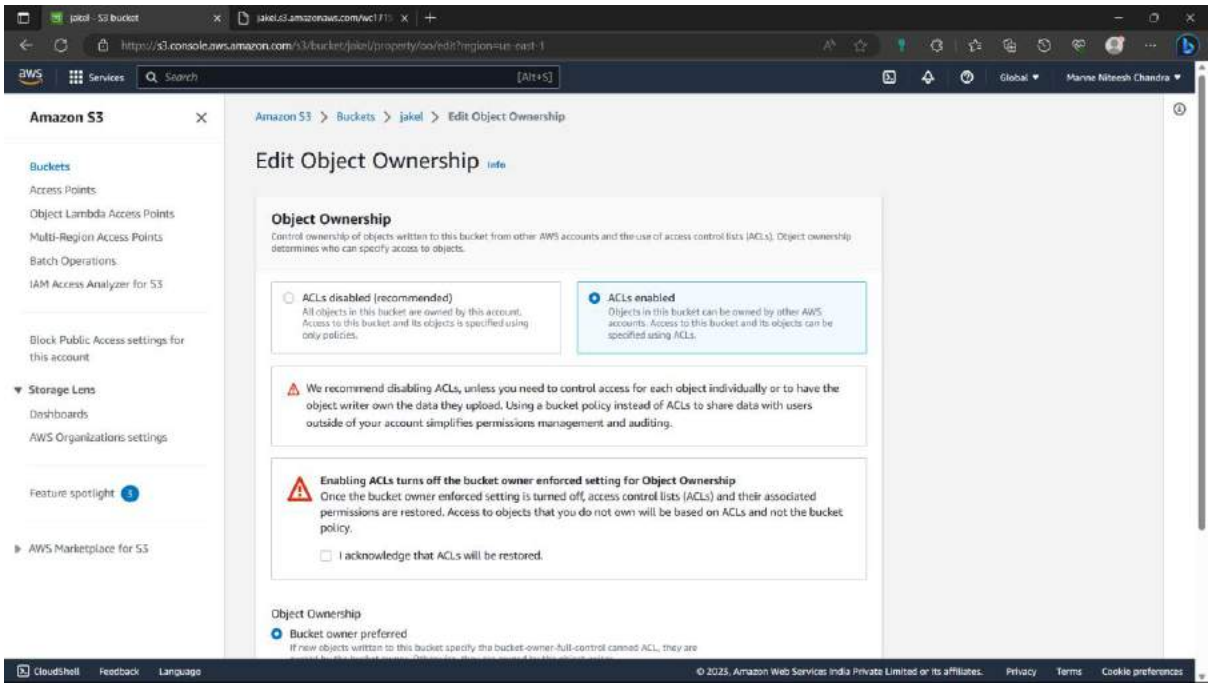
85)It shows error because we didn’t assign any permissions to the file. Now we assign permission to this file.



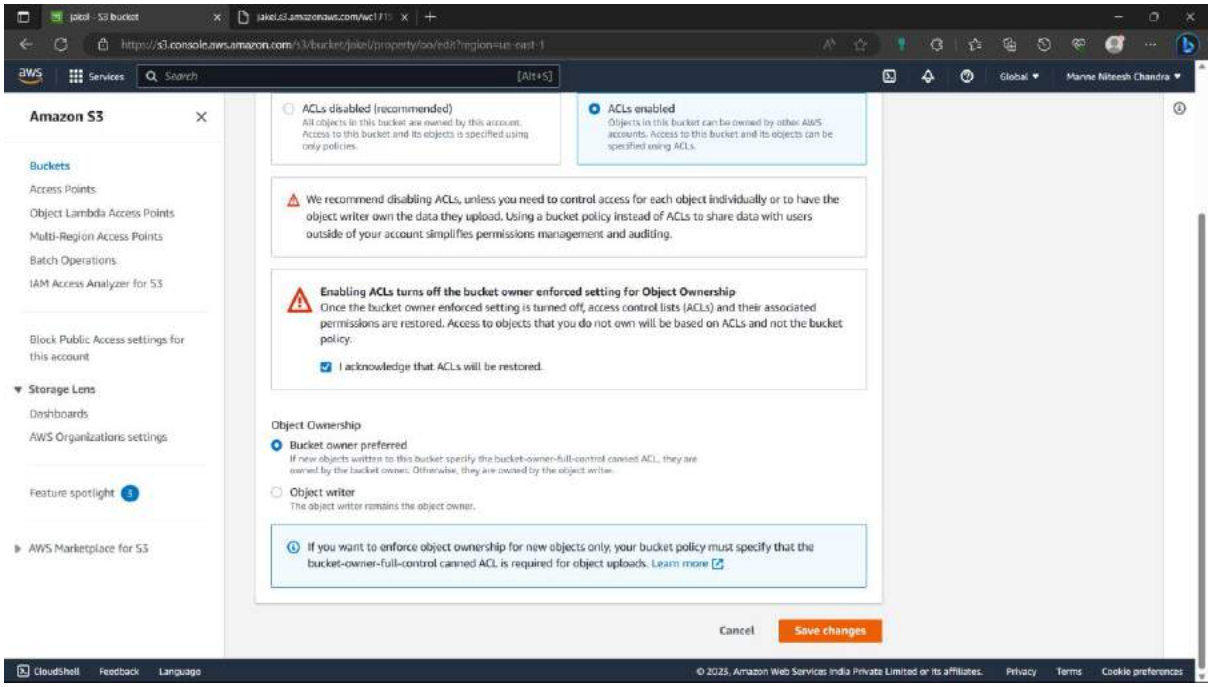
86)Navigate to the permission option in the file. Click on “bucket owner enforced” option show in blue colour.



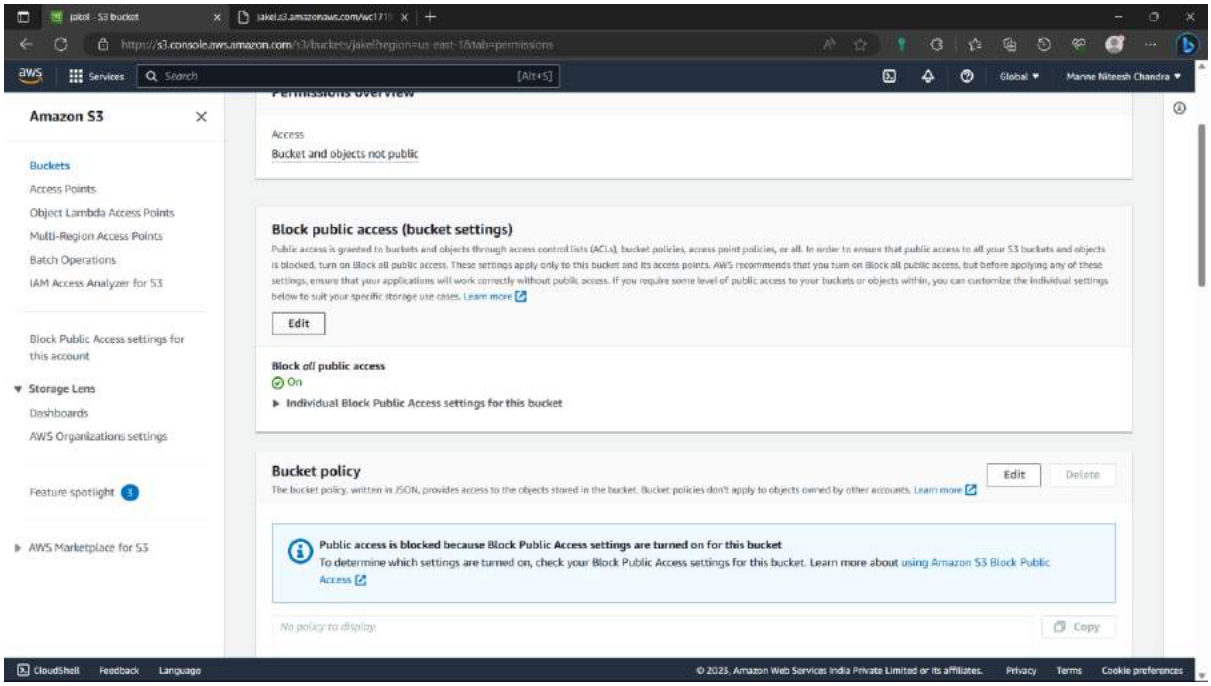
87)It will navigate to Object ownership. Enable “ACLs enabled”.



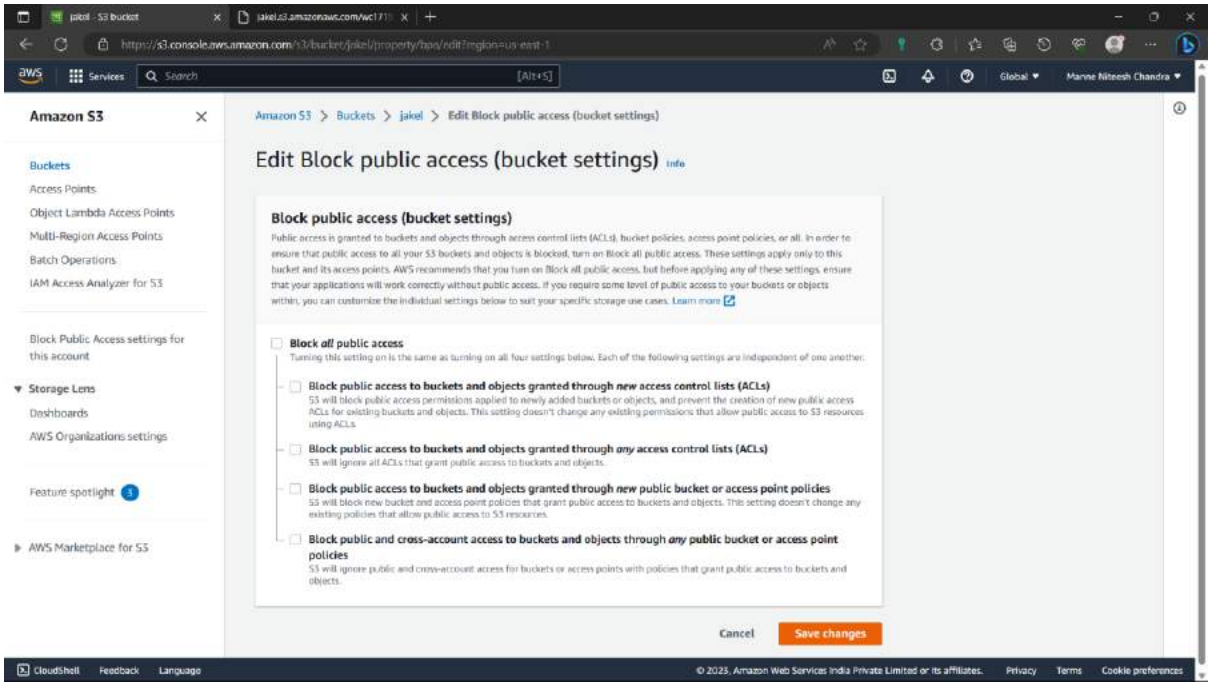
88)Enable the “I acknowledge that ACLs will be restored” and click on “save changes” option.



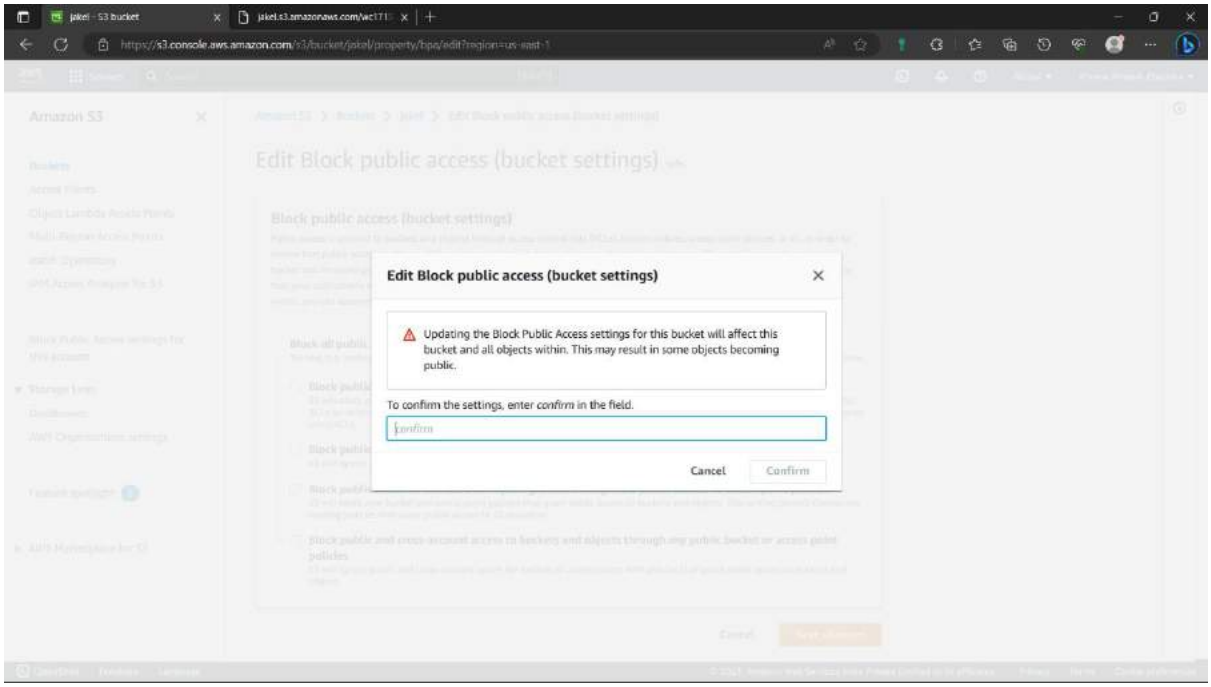
89)Now the bucket became public. Click on edit option for “Block public access”.



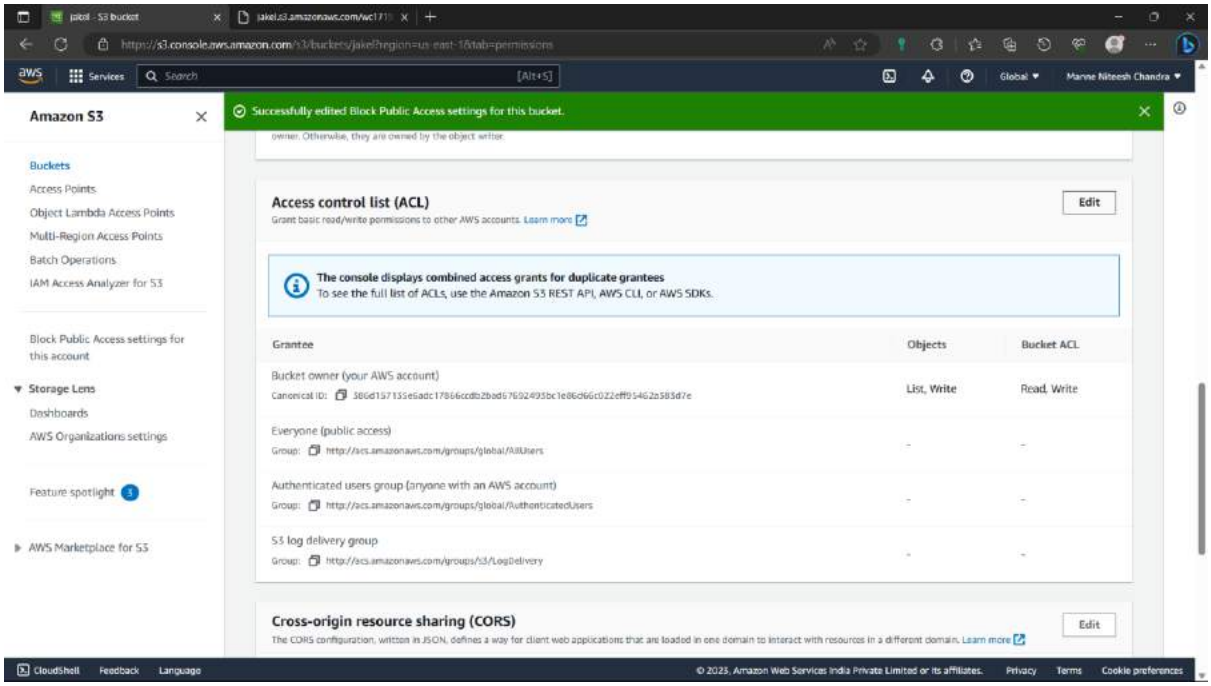
90)Uncheck the “Block all public access” option and click on “save changes”.



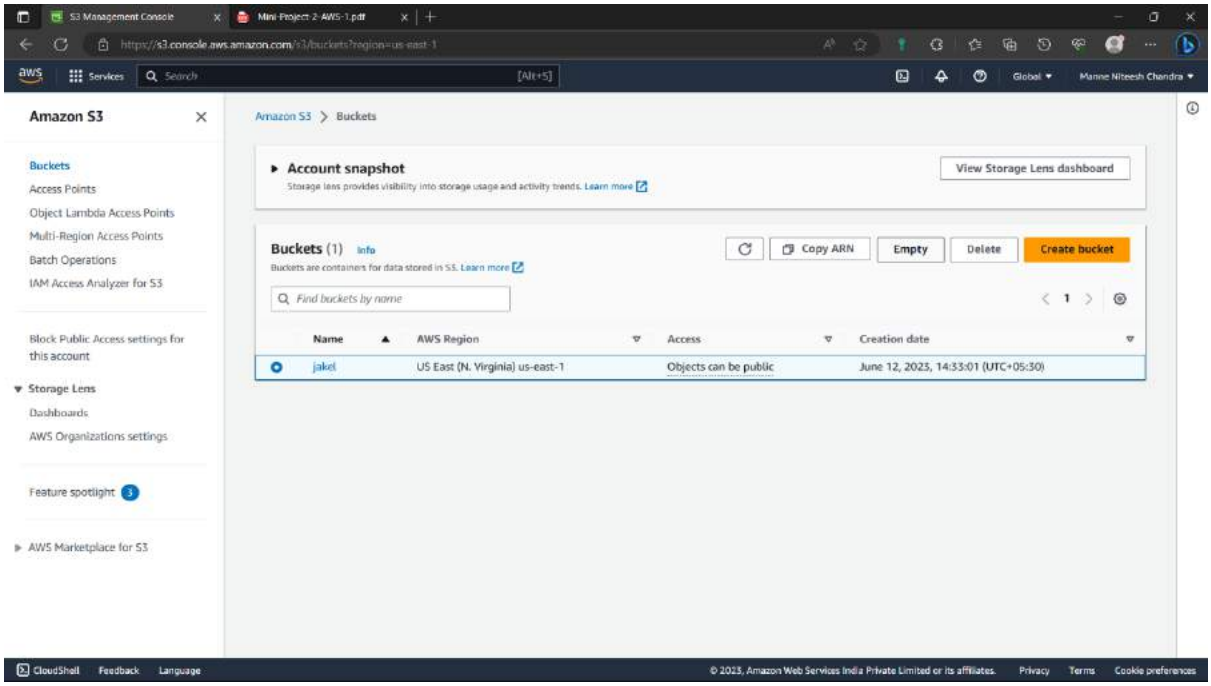
91)Type “confirm” and click the confirm button.



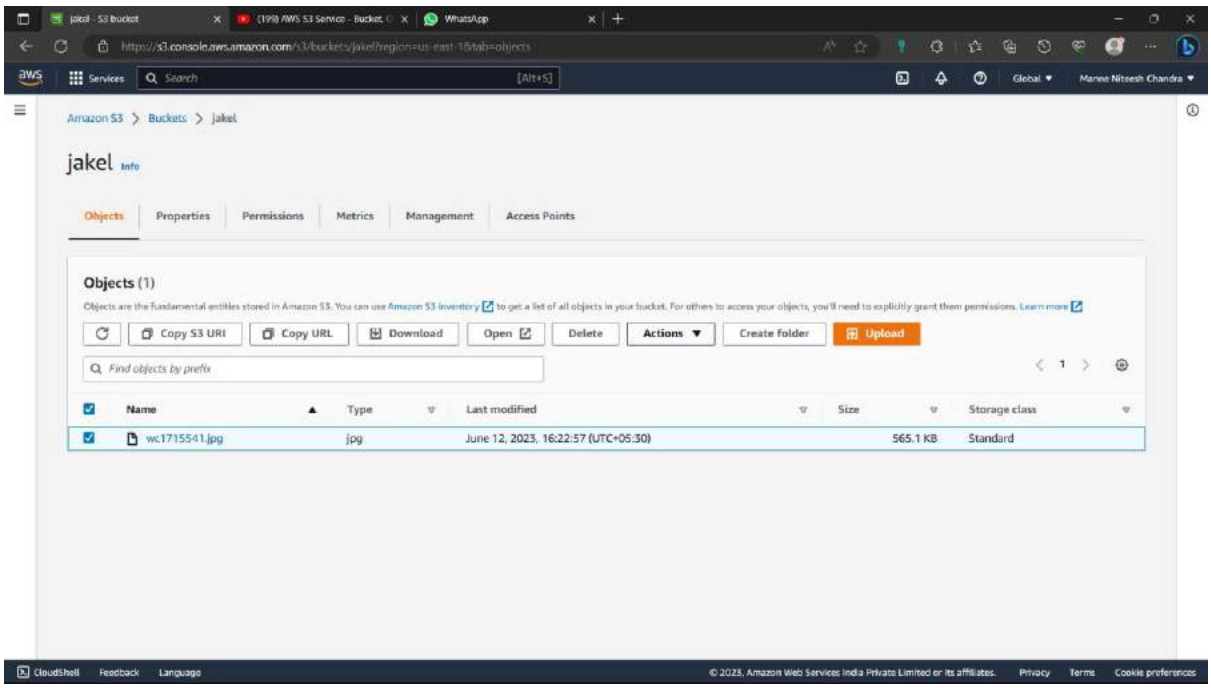
92)Now the data can be accessed by public.



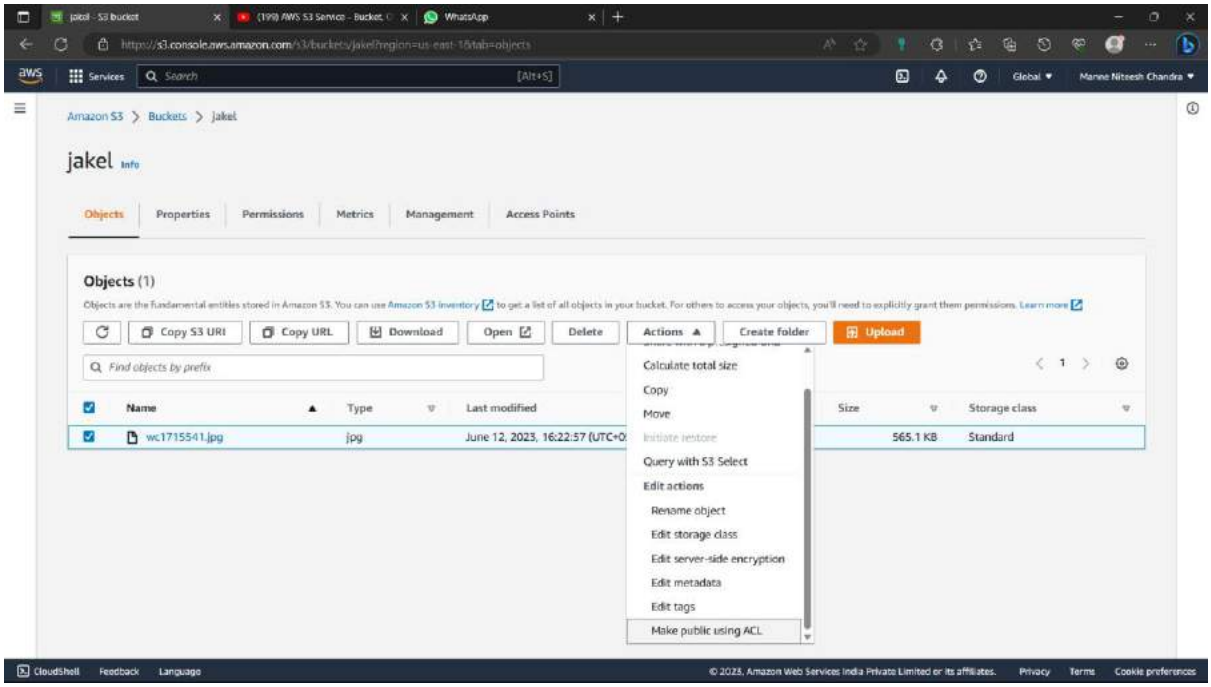
93)Now go back and open the created S3 bucket.



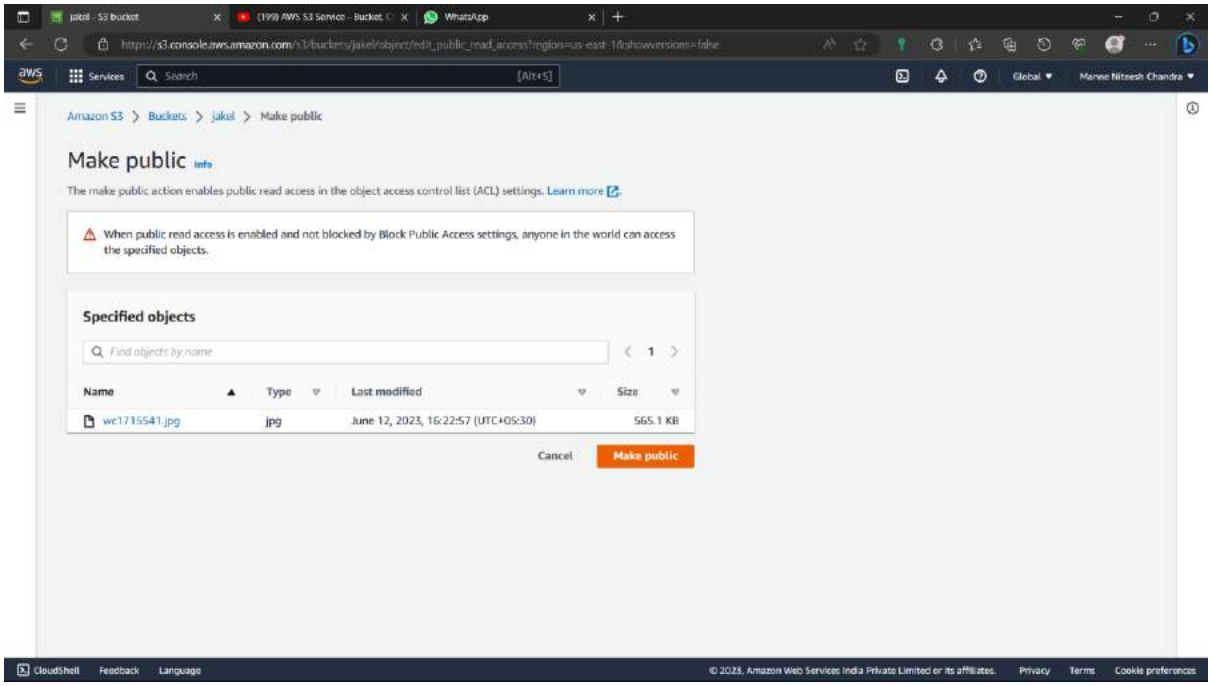
94)Select the file i.e., uploaded and click on actions.



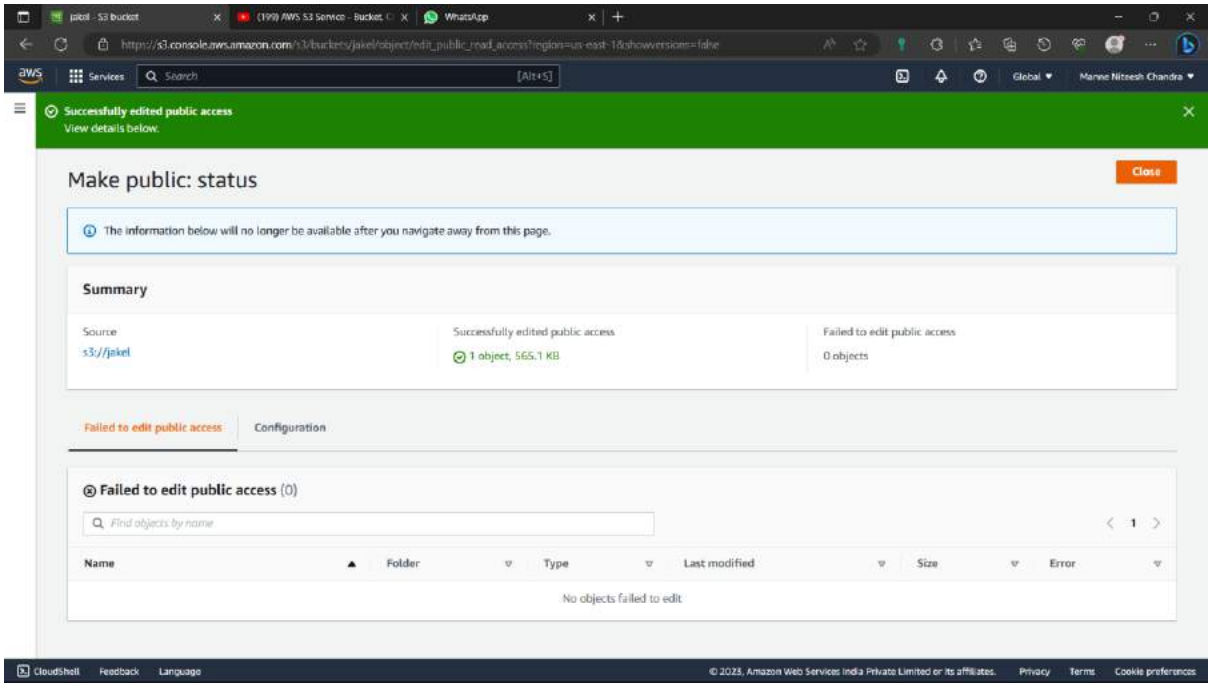
95)Select the”Make public using ACL” option in the bottom.



96)Click on “make public”.

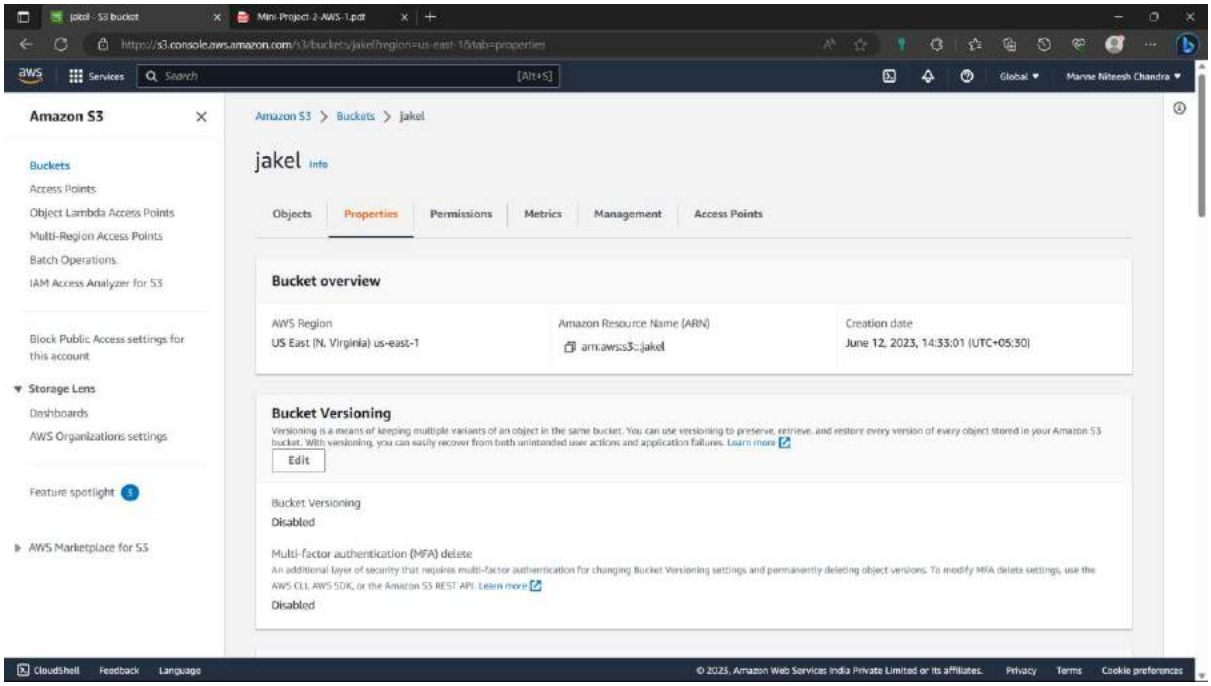


97)Now the file also became public.

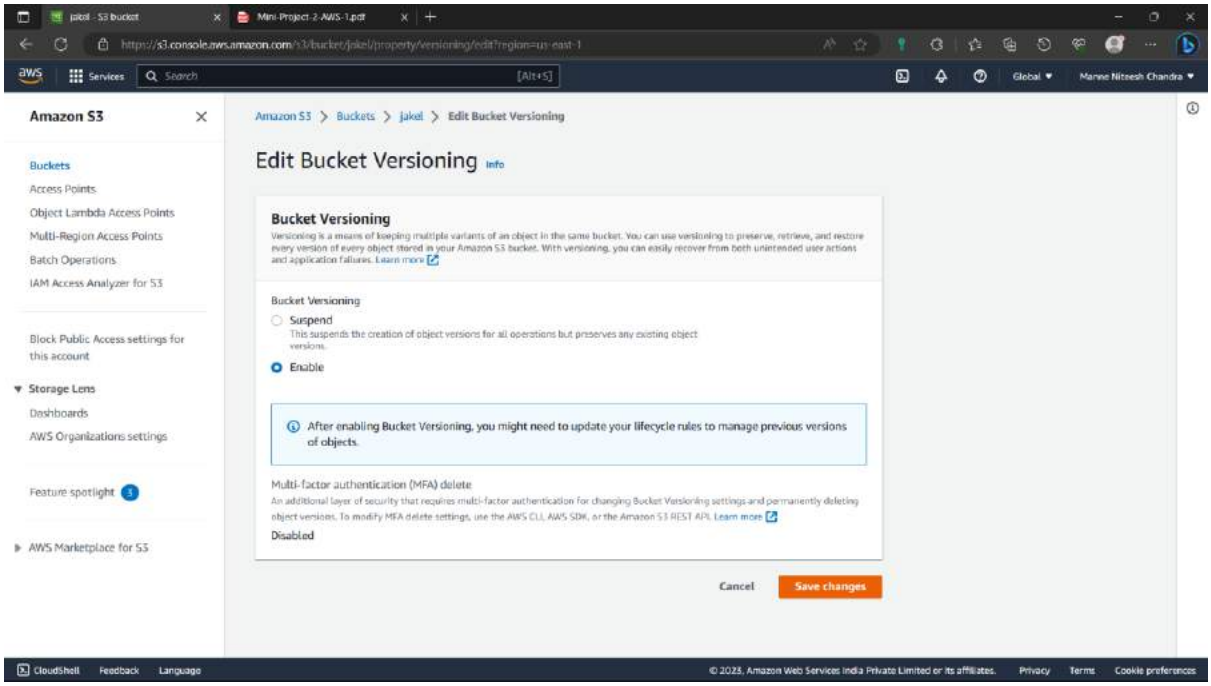


98)Now refresh the “Object URL” page we can see the image i.e., uploaded in S3 bucket.

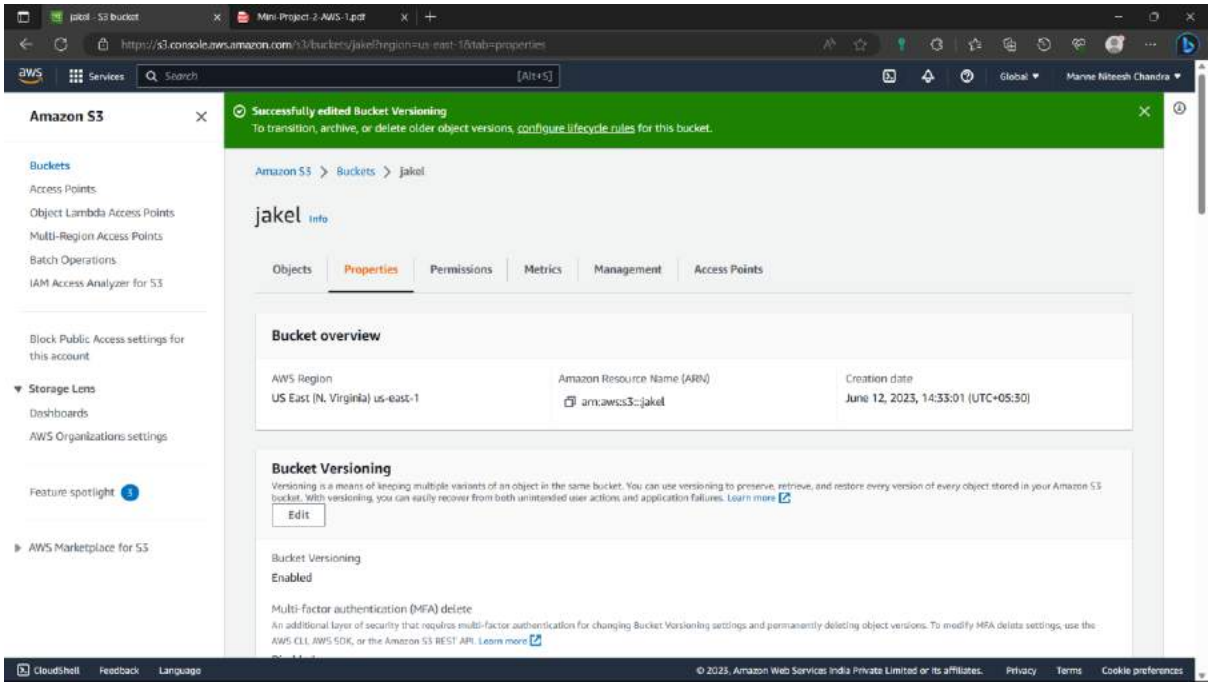
99)Now go to the properties of your created S3 bucket. Click on edit option of “Bucket Versioning”.



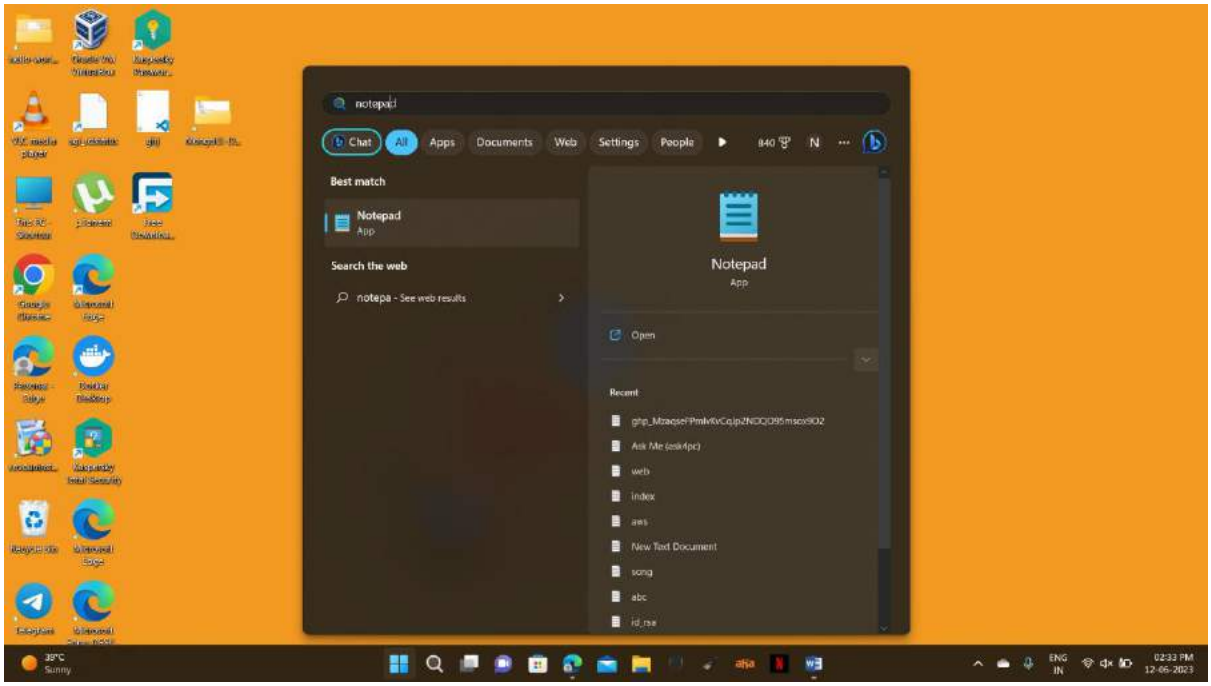
100)Select the “Enable” option and click on save changes.



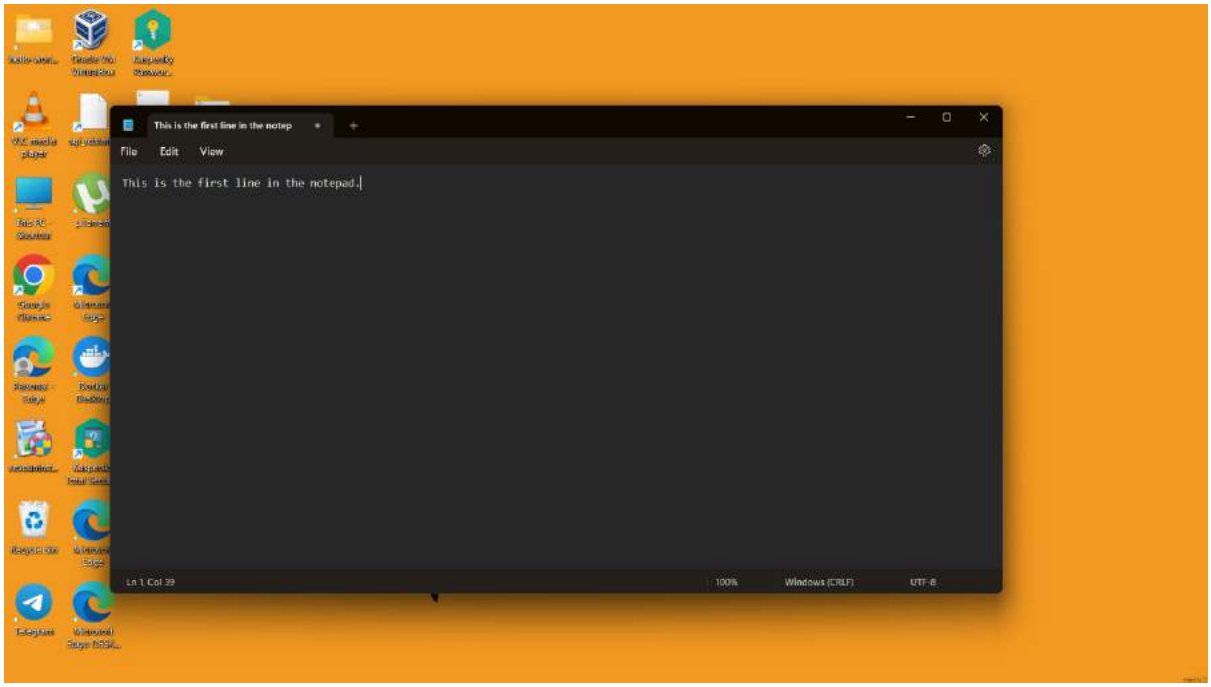
101)Now the bucket versioning is successfully edited.



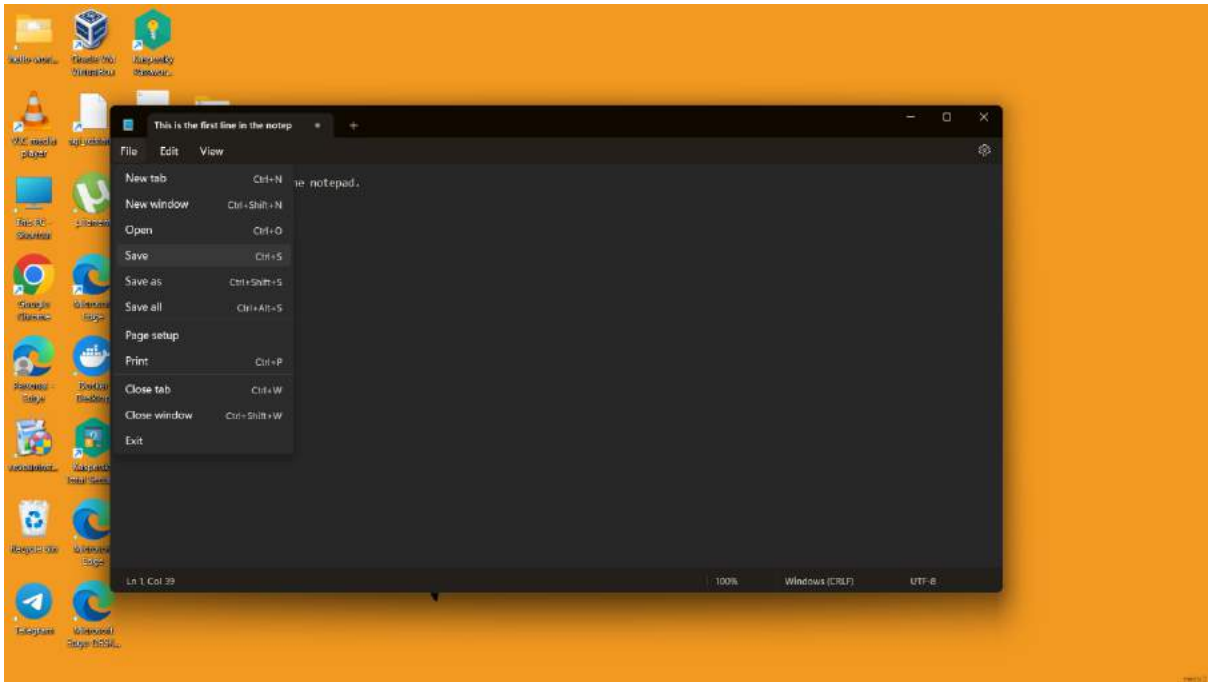
102)Now go to your local windows and open notepad.



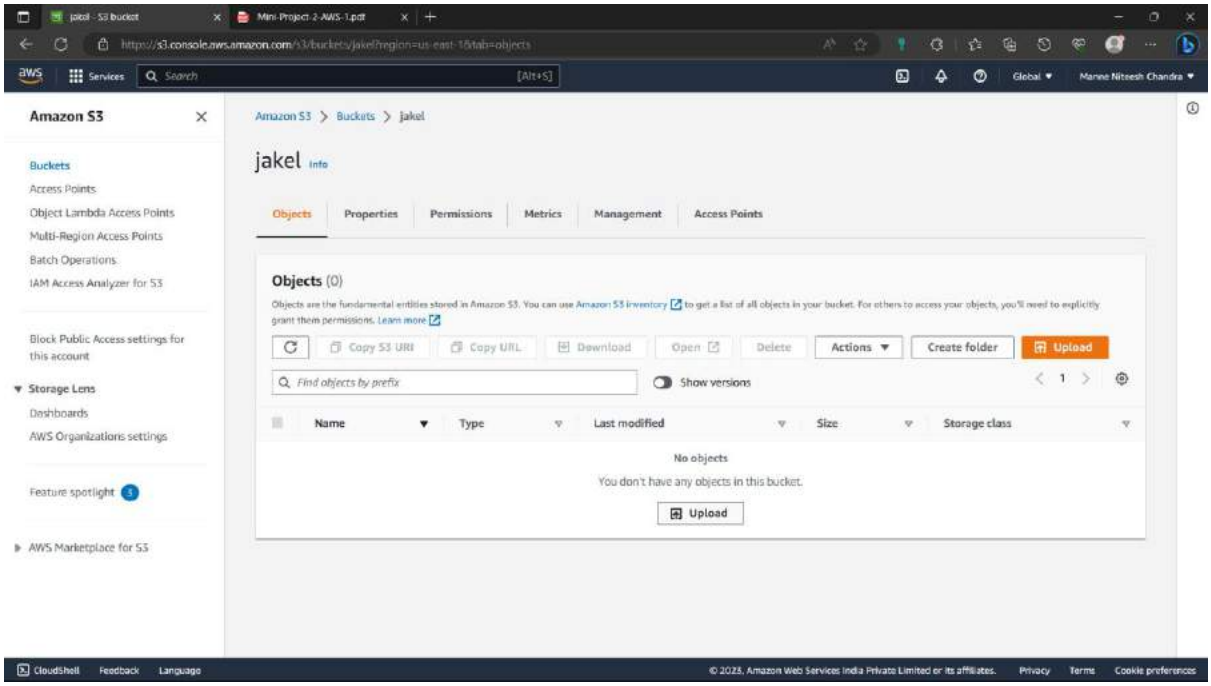
103)Type some content(random) in the notepad.



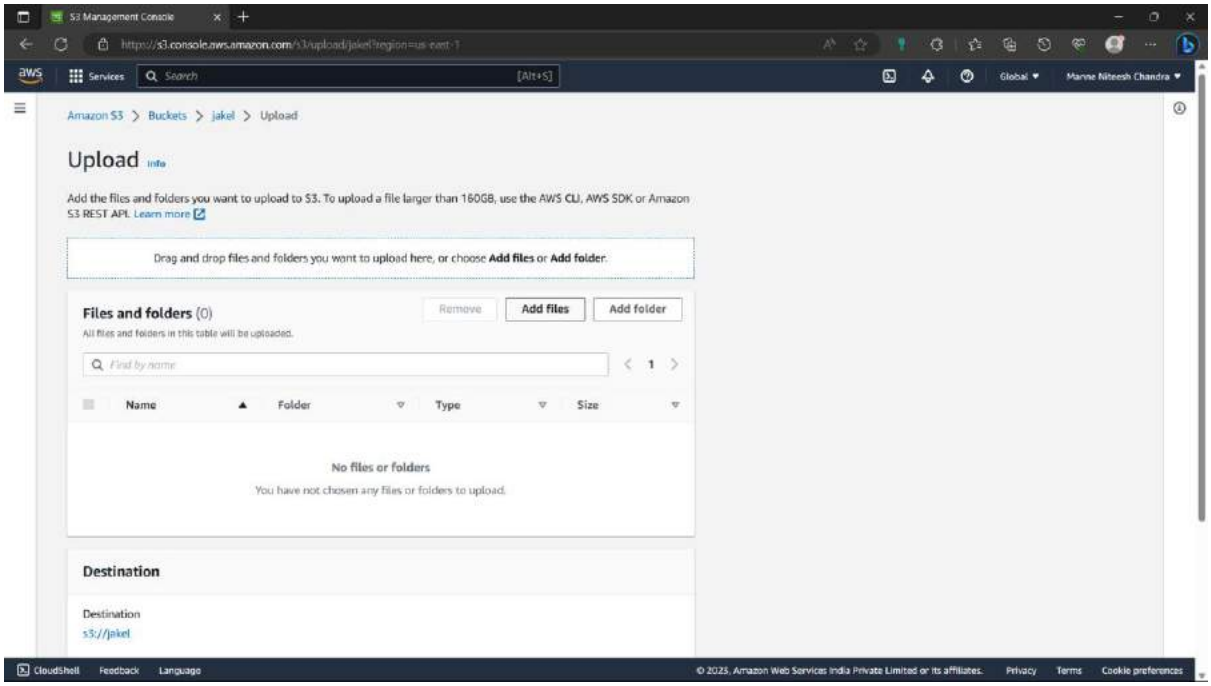
104)Save it as a text file in the windows.



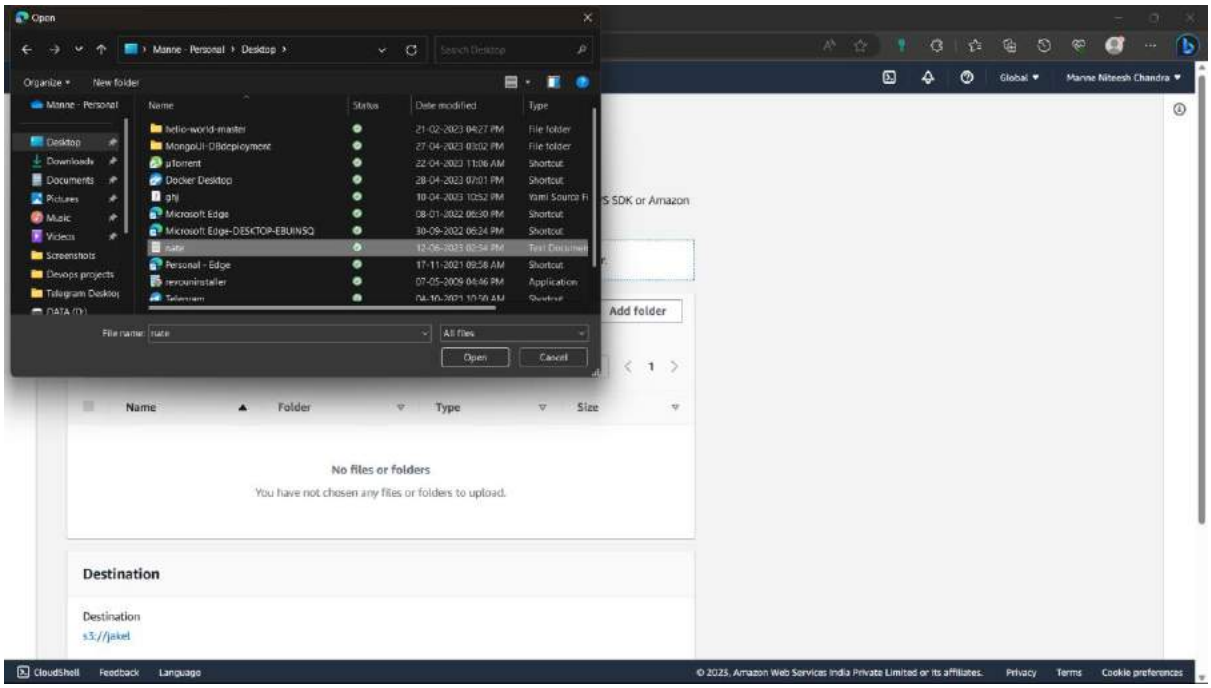
105)Go to S3 bucket you created in AWS account and click on upload.



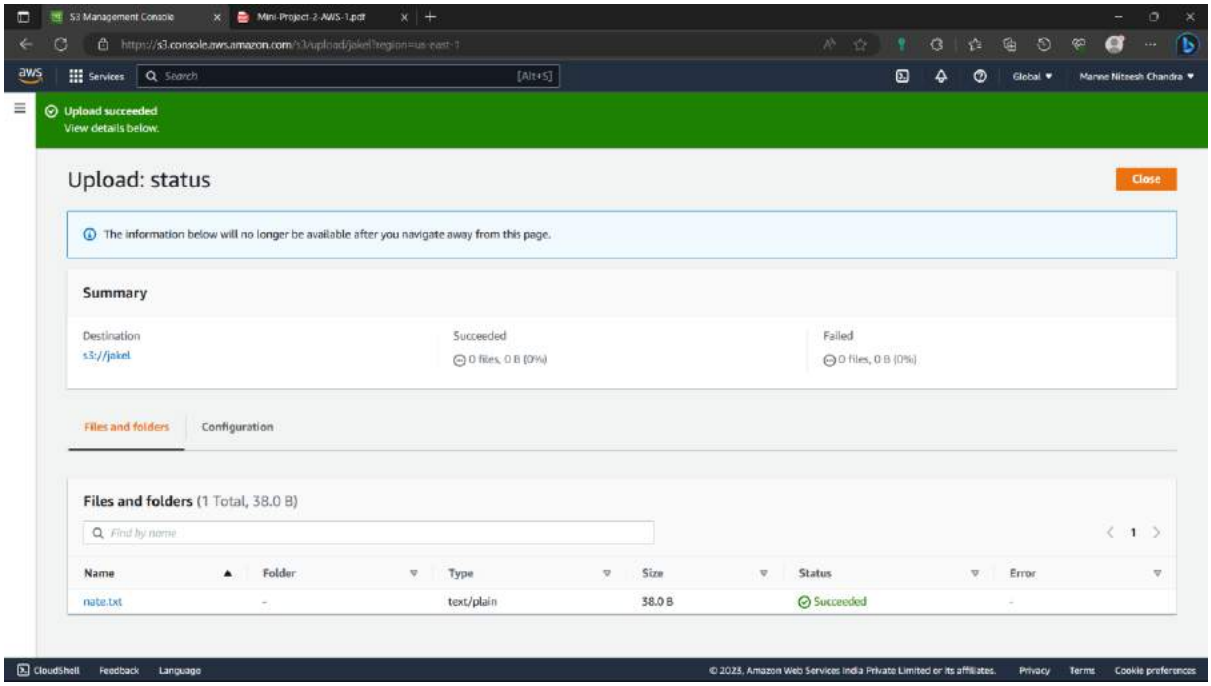
106)Click on add file.



107)A new window is appeared with local files, Select the text file you created.

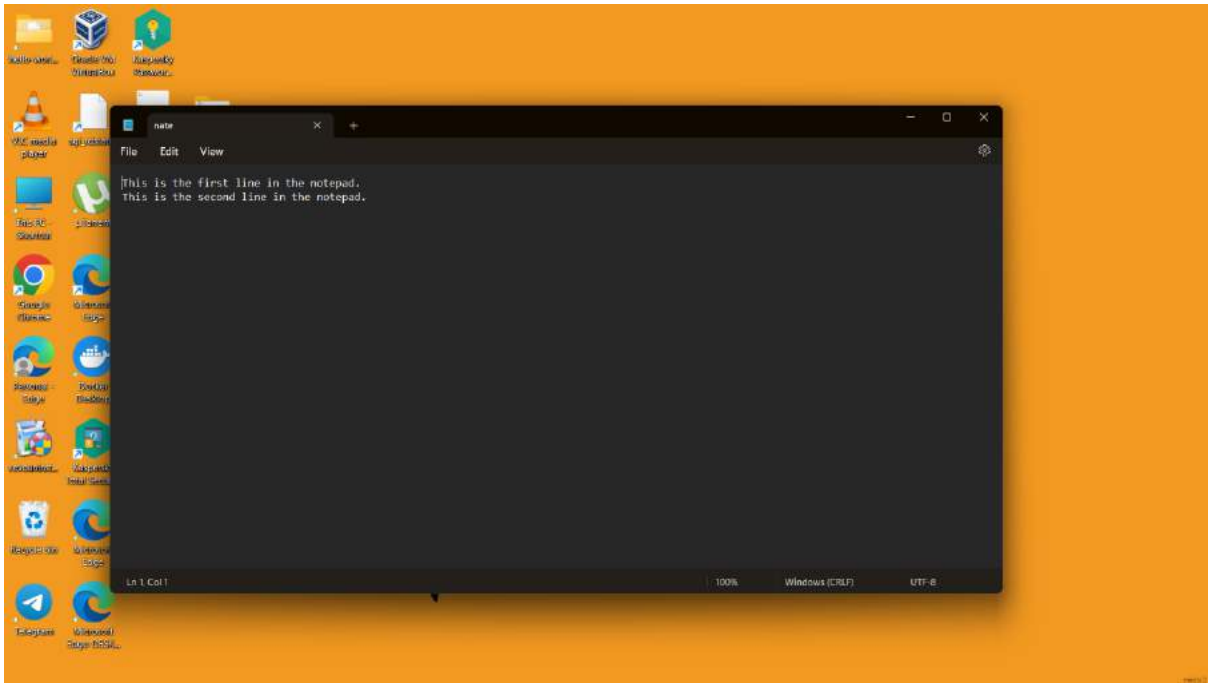


108)The text file is uploaded.

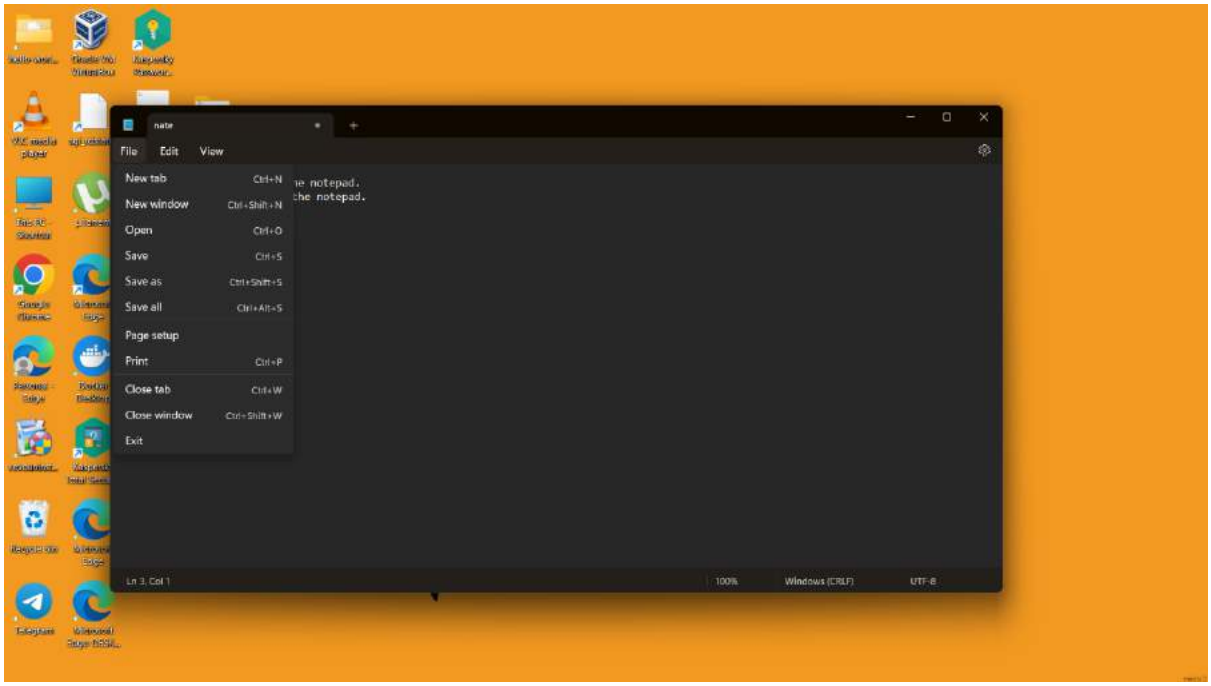


109)Now open the same text file again in the local host.

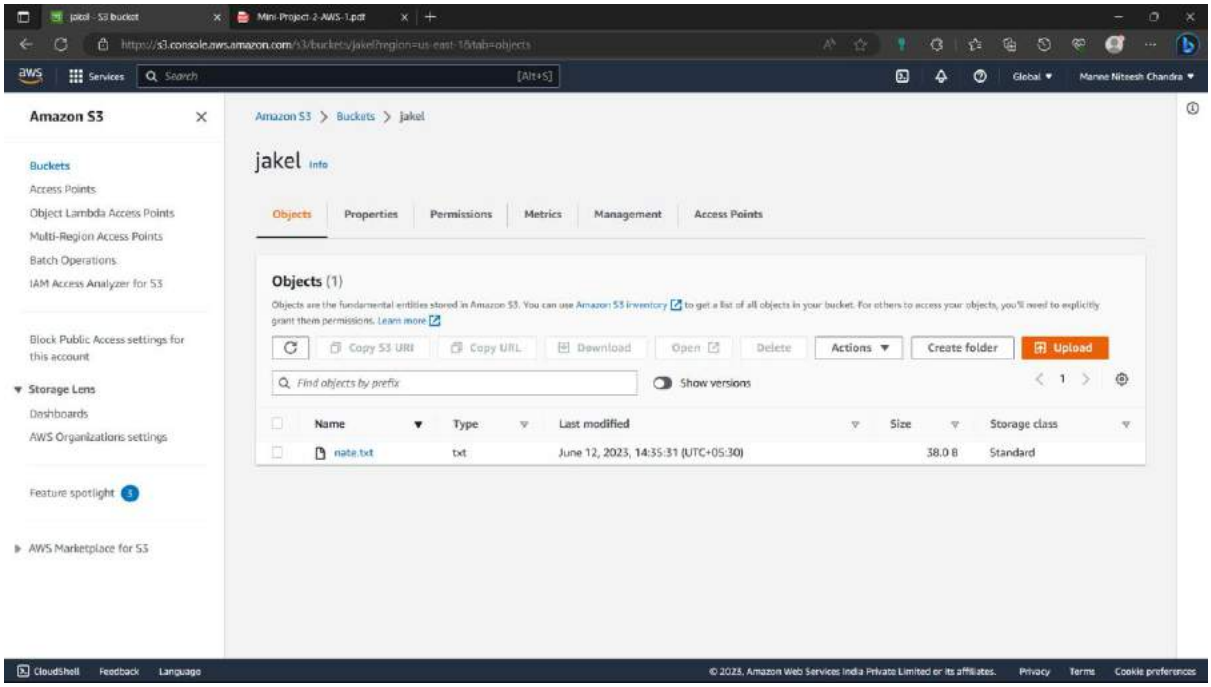
110)Before adding extra content save a copy of the current file in another path for checking purpose. Now add extra content to the file.



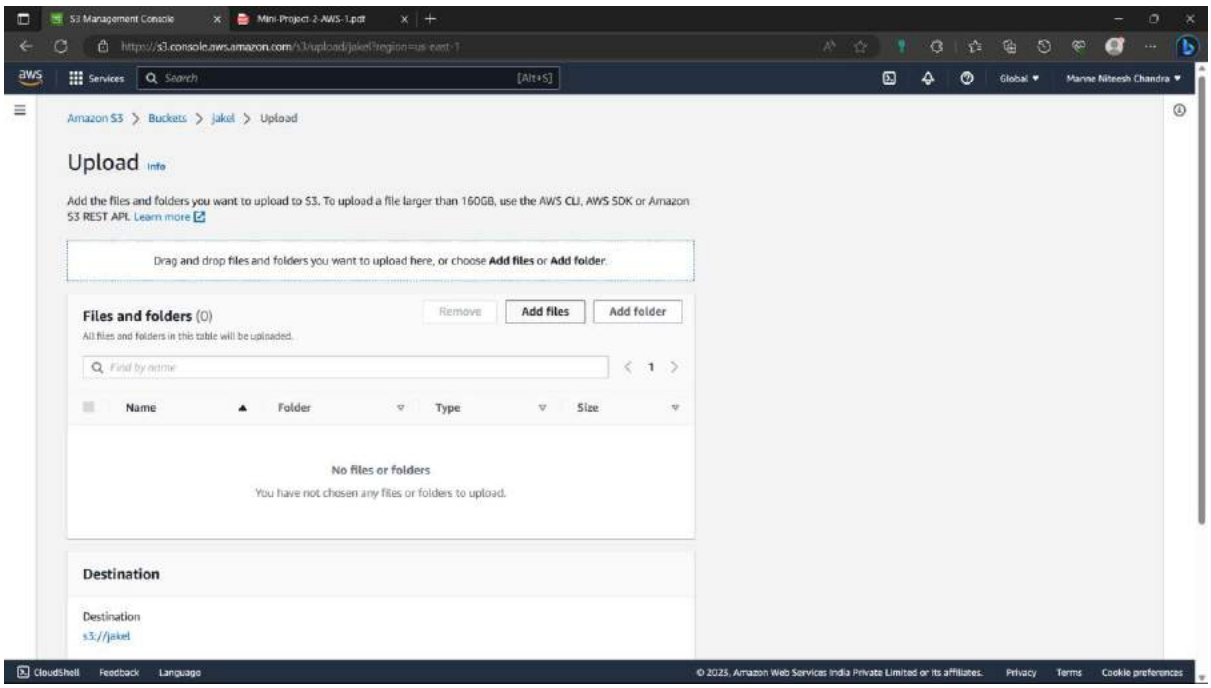
111)Don’t change the text file name and save it.



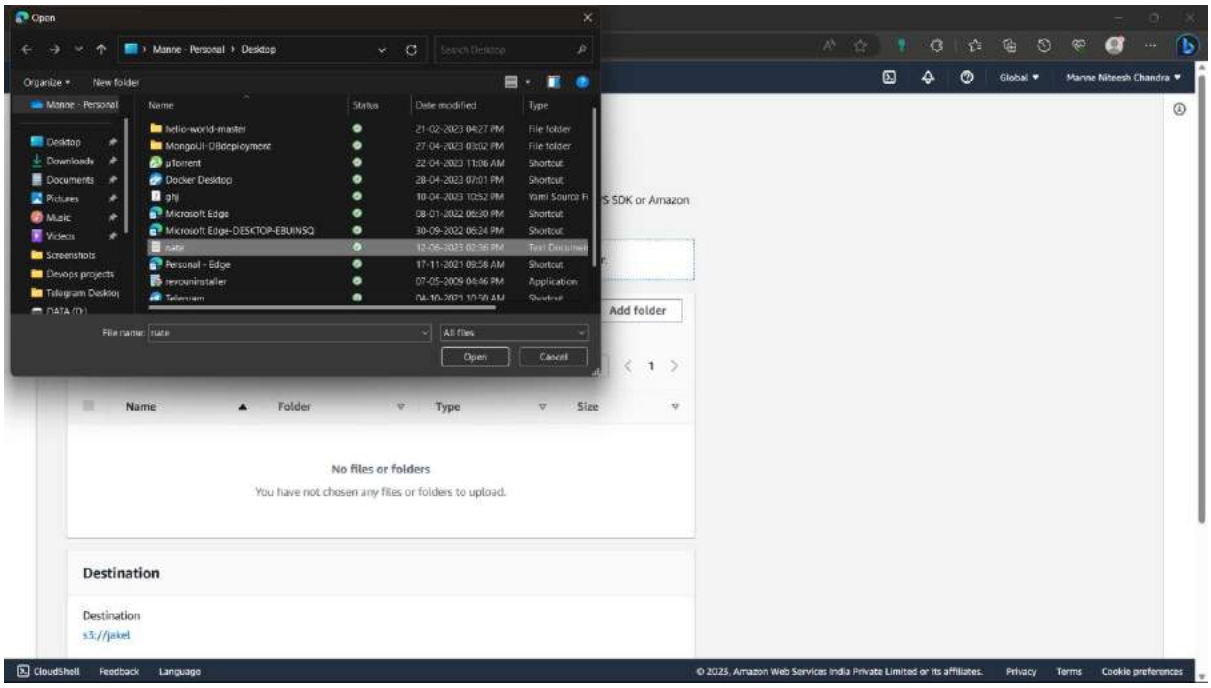
112)Now upload the file in the S3 bucket. Go to the S3 bucket you created and click on upload option.



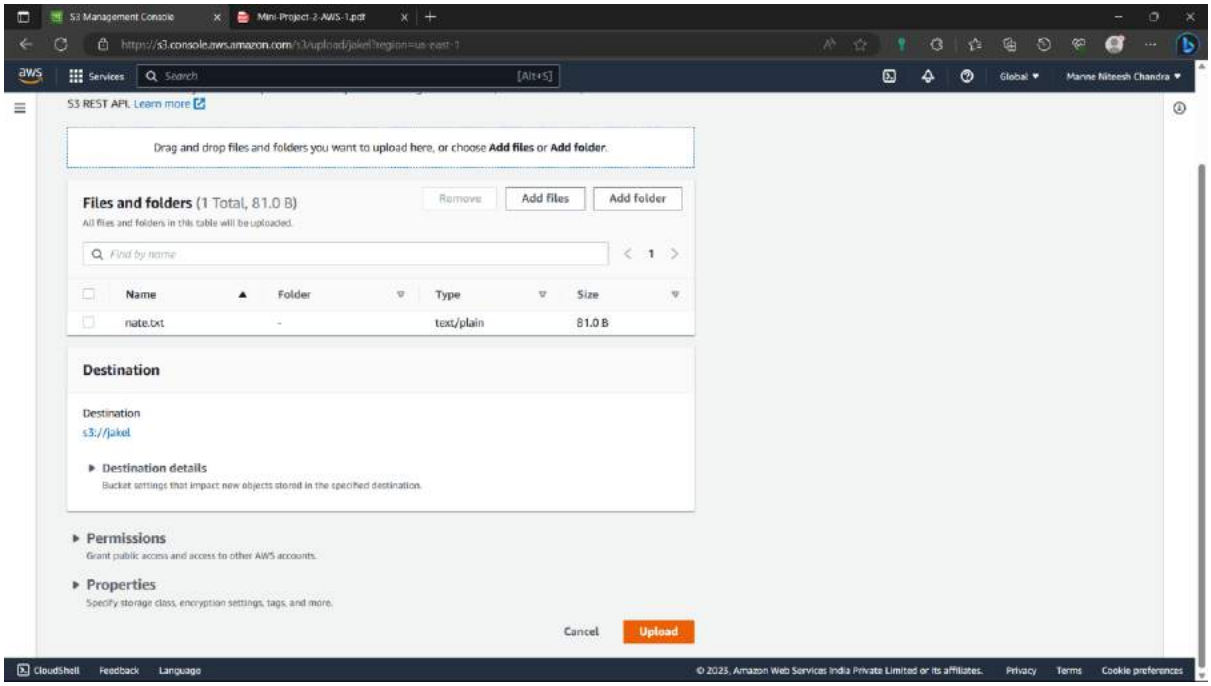
113)Click “Add file” option.



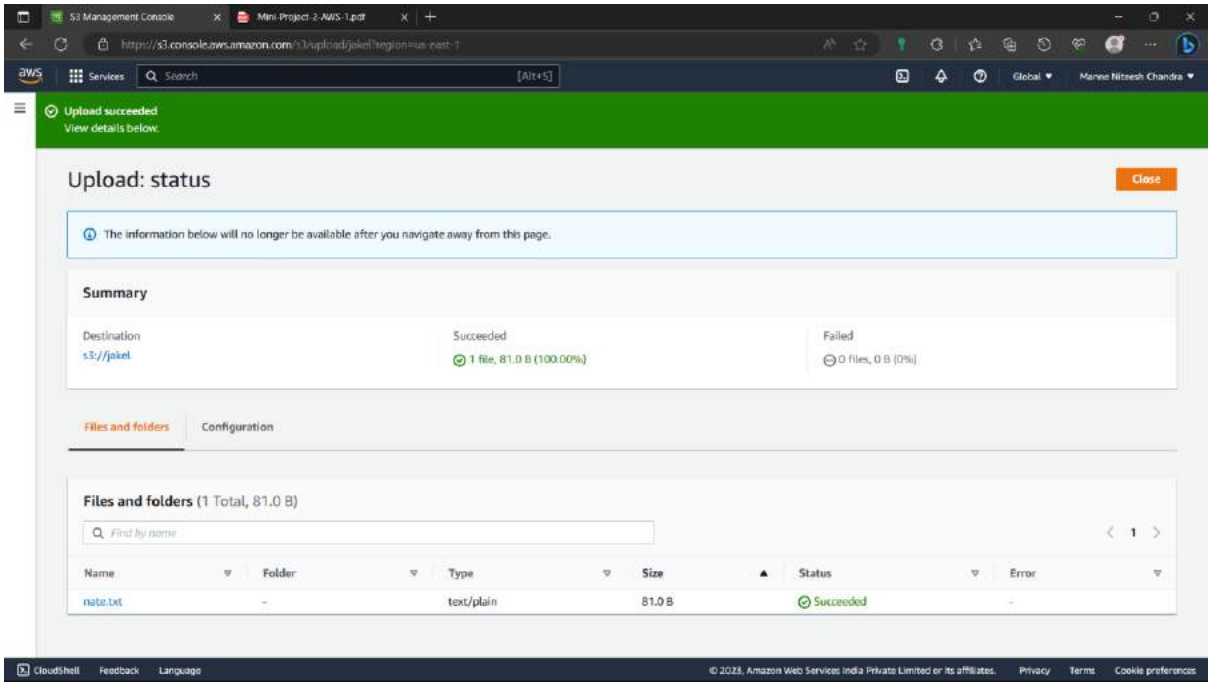
114)Now a new window with local files will be opened select the file i.e., content is upadated.



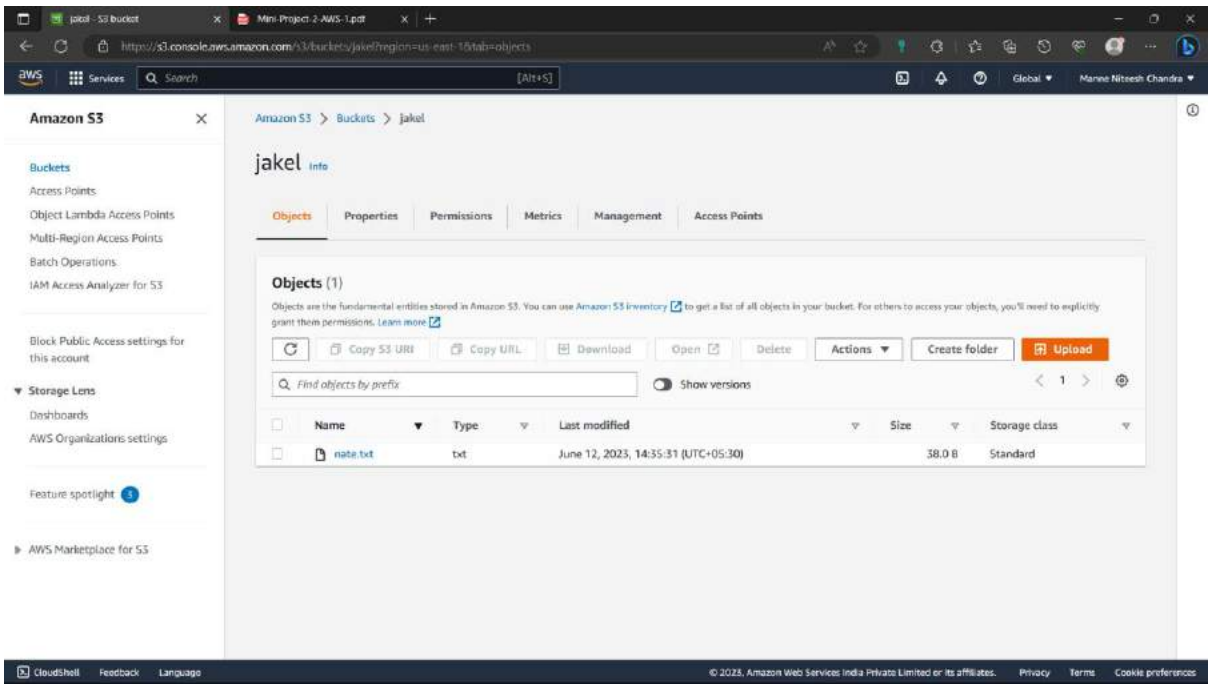
115)Click on “upload” option.



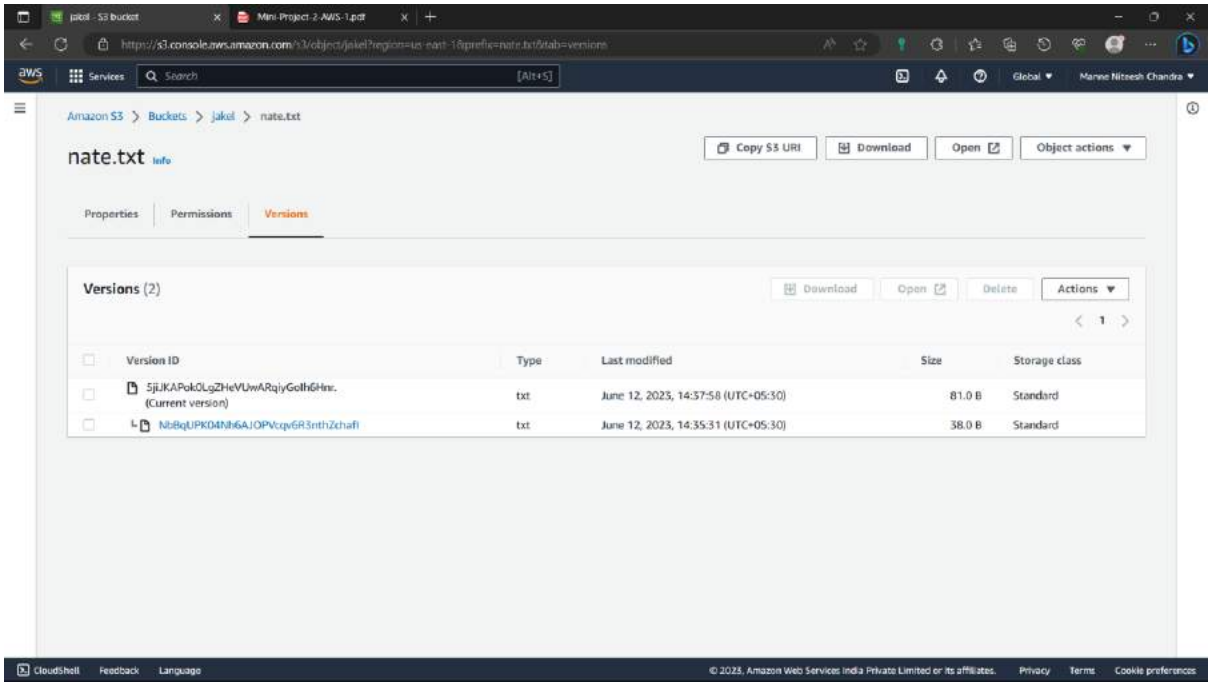
116)Now the files has been uploaded successfully.



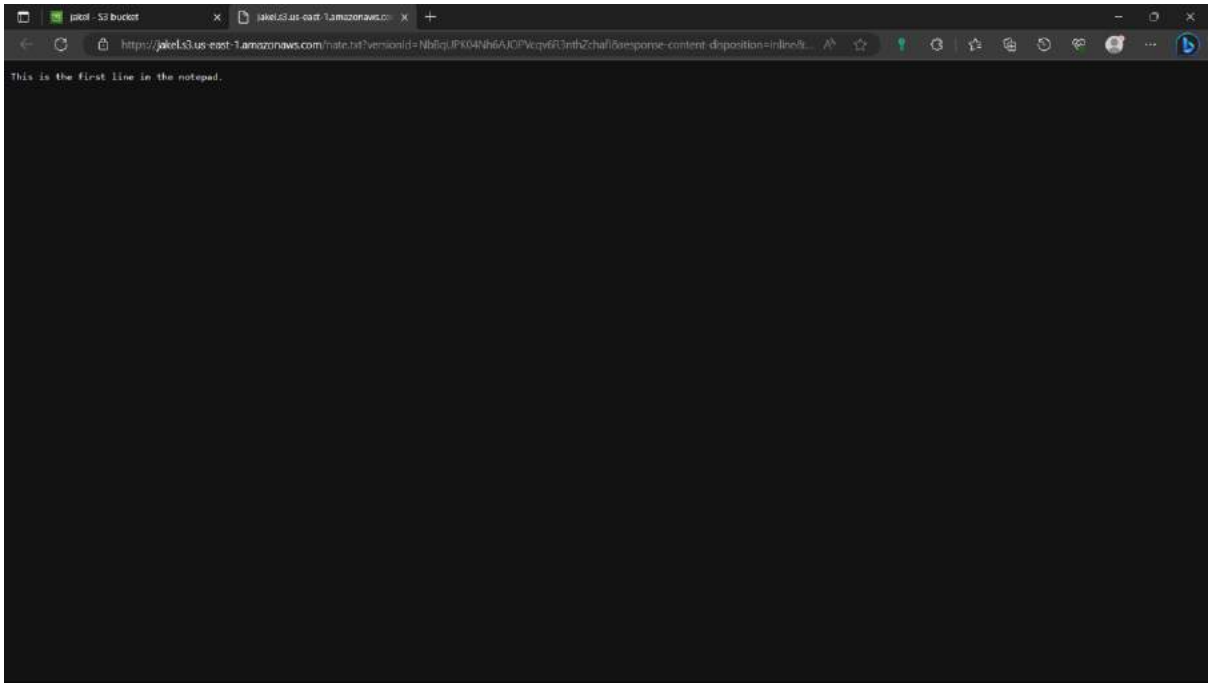
117)Rather than the files showing separately it shows only the current updated file.



118)Now click on the “show older version”. It will show the older file i.e., first updated file. the files are arranged like this because of bucket versioning option. The current version means the file which is currently is updated and using.

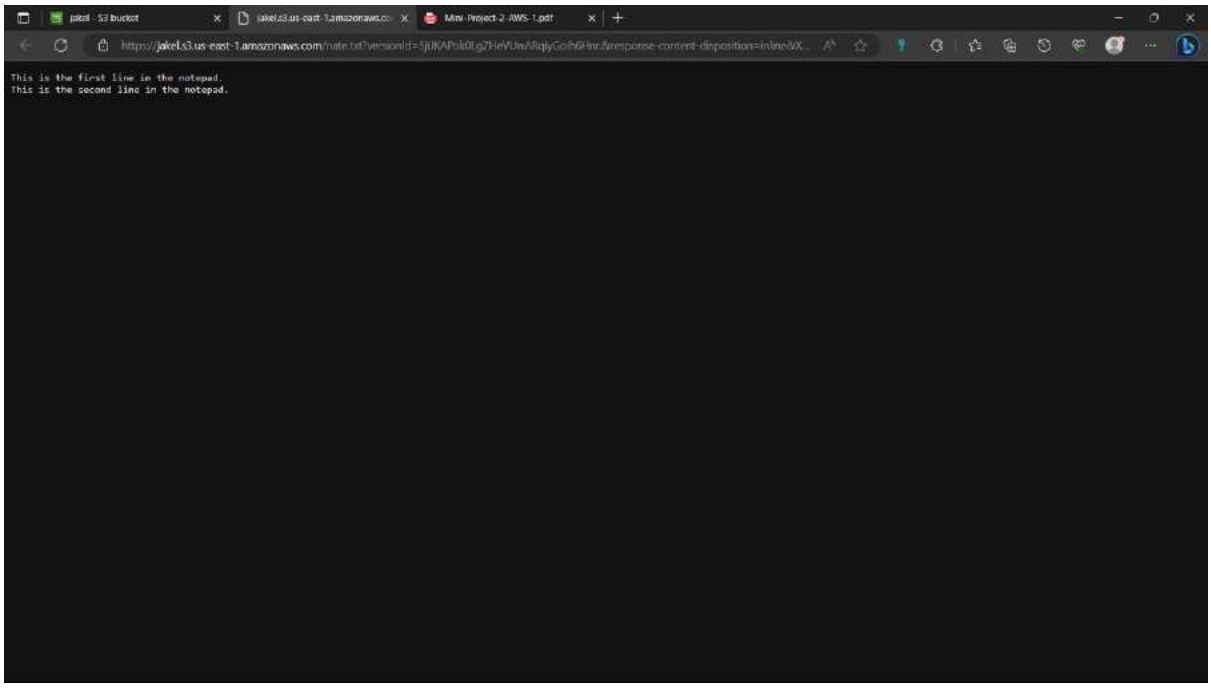


119)Now check the contents of both the files. Using the object URL. The first uploaded file content is shown below. Check it with your local text file where you save the copy in a different path. If you get error while openinh this file in new window repeat the steps (86-98).

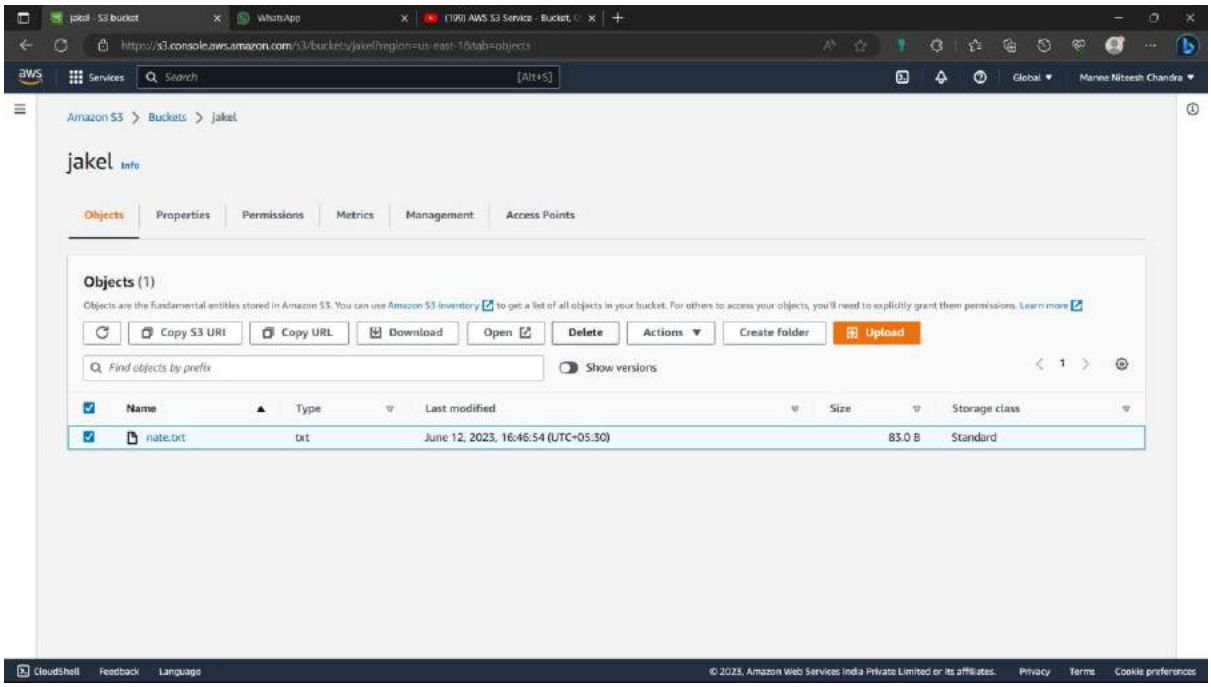


120)Now check the contents of the second uploaded file.

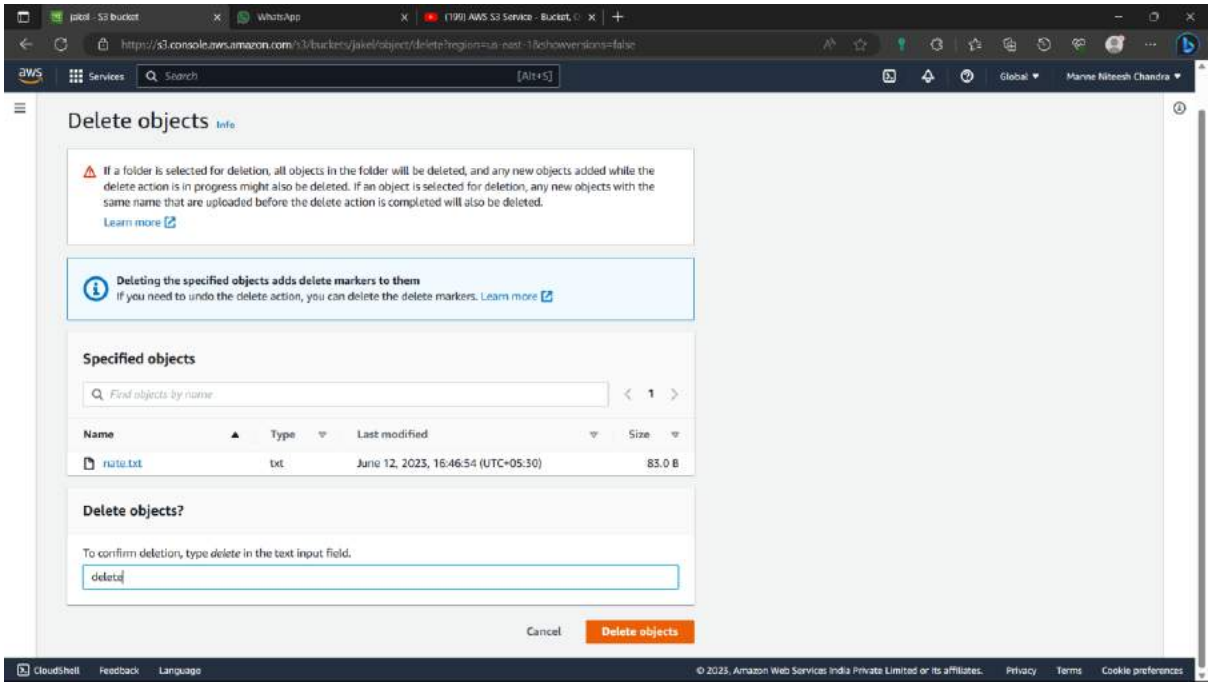
Now



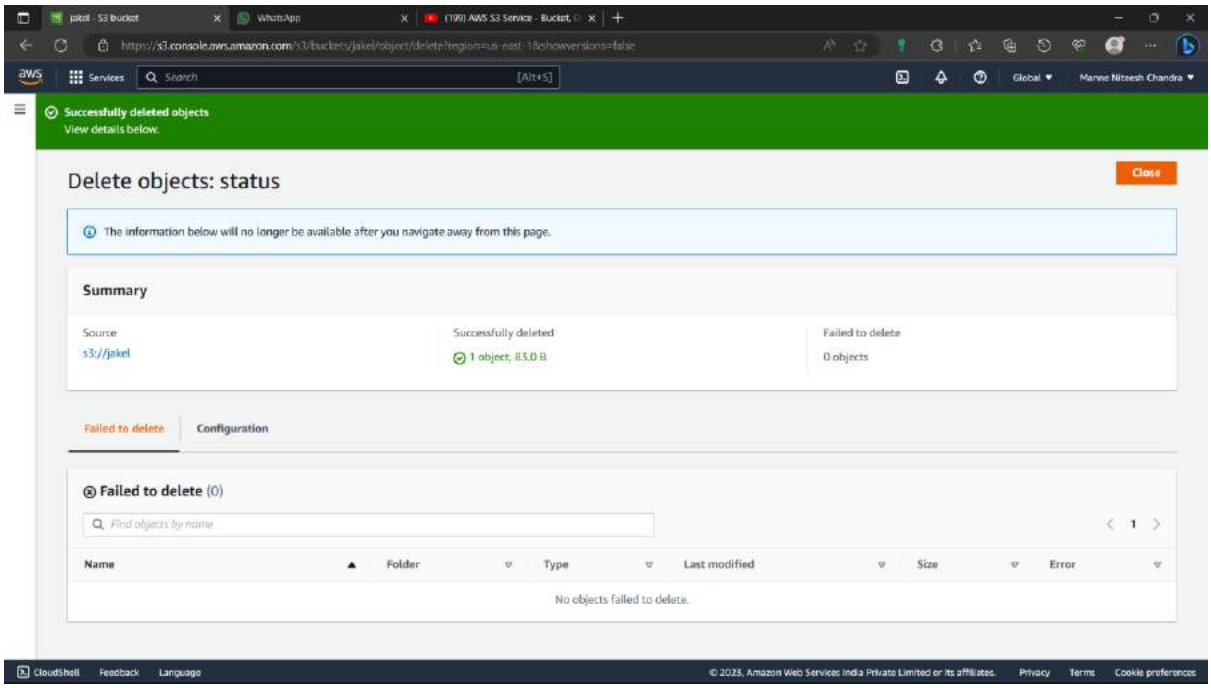
121)Now go to object of file in the S3 bucket. Select the file created and click on delete option.



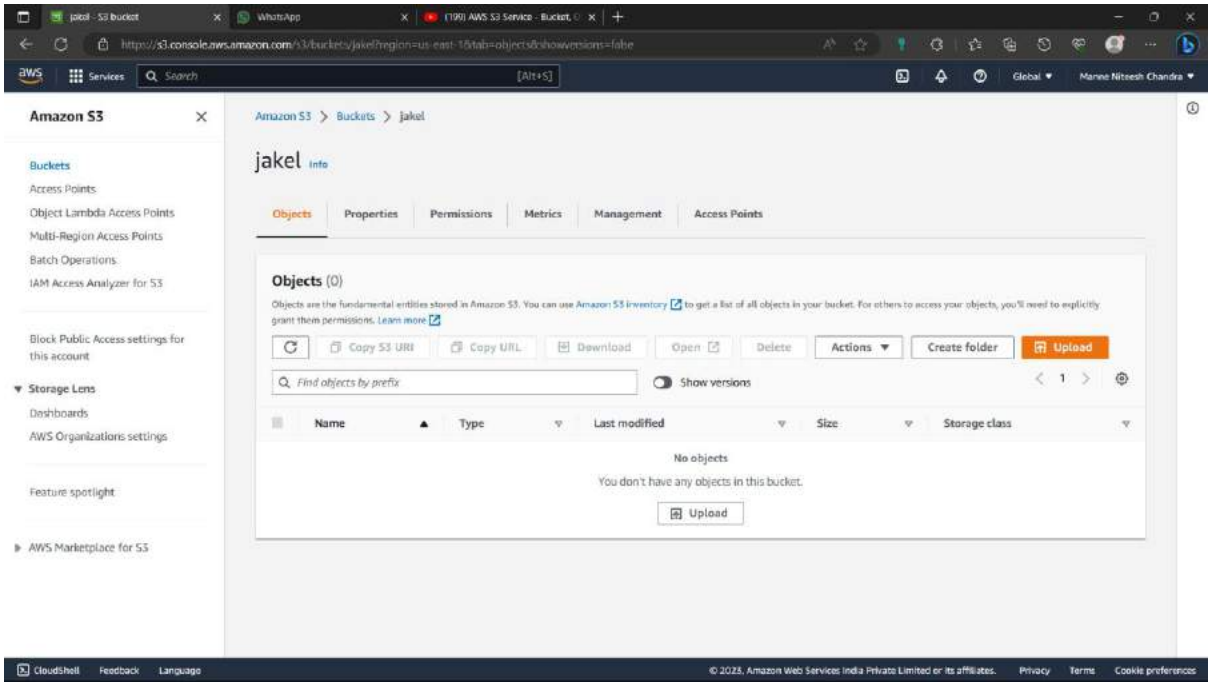
122)Type “delete” and click on “delete object” option.



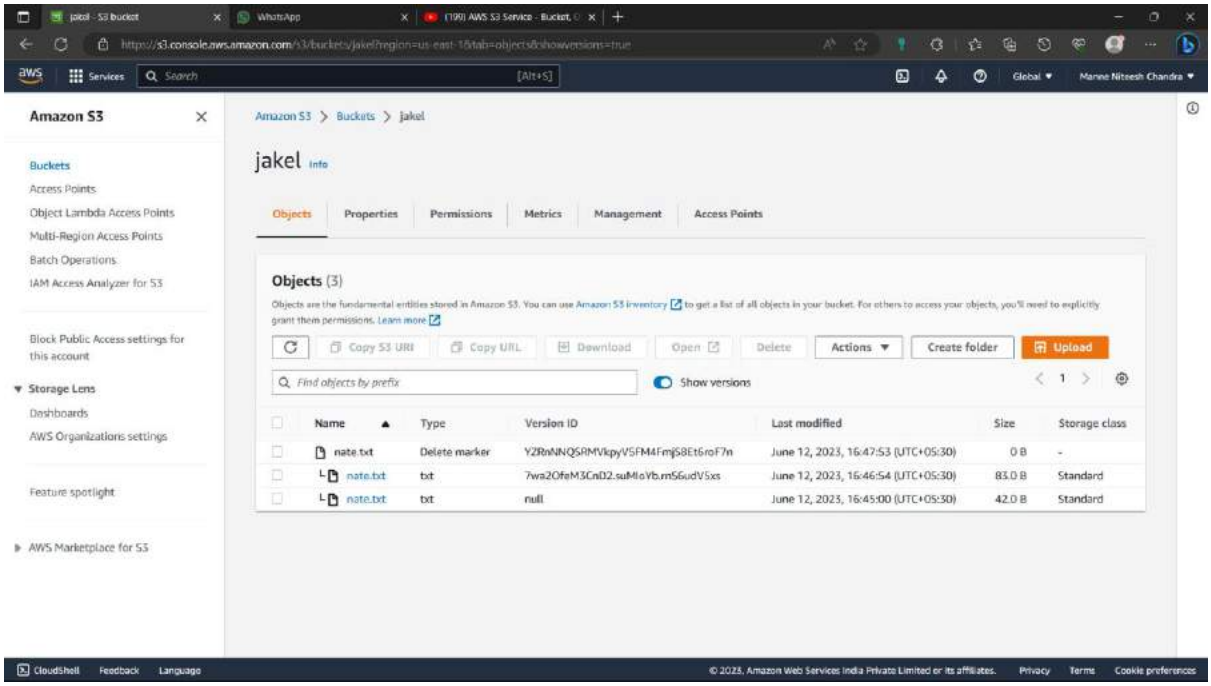
123)Now the files are deleted.



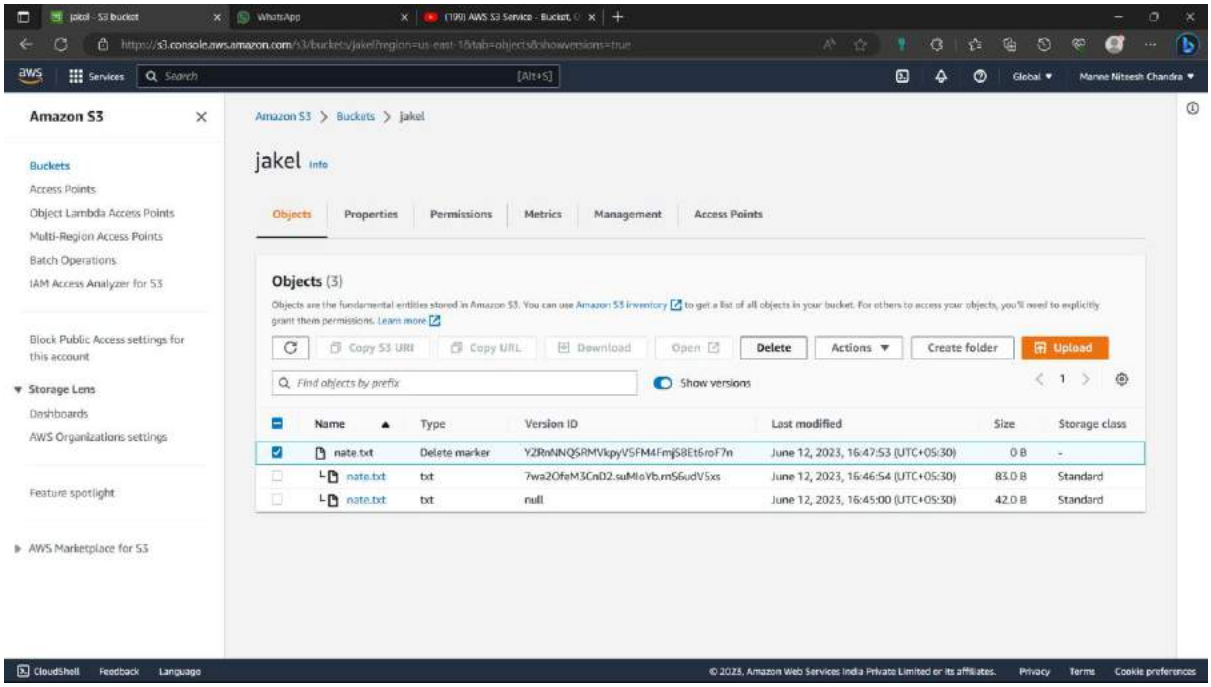
124)Even though the files are deleted because of bucket versioning they won't be deleted permanently. Go to objects in the S3 bucket you created. Click on "Show versions".



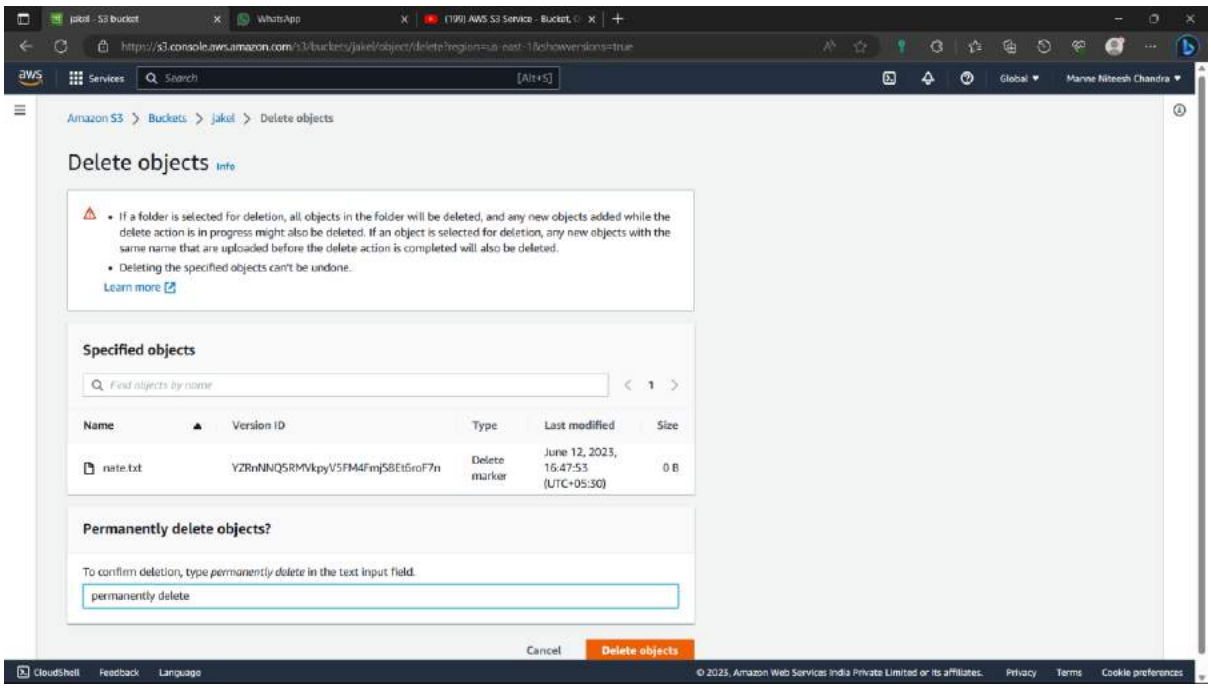
125)Now we can see a marker file(Copy of the original file) and the two text files.



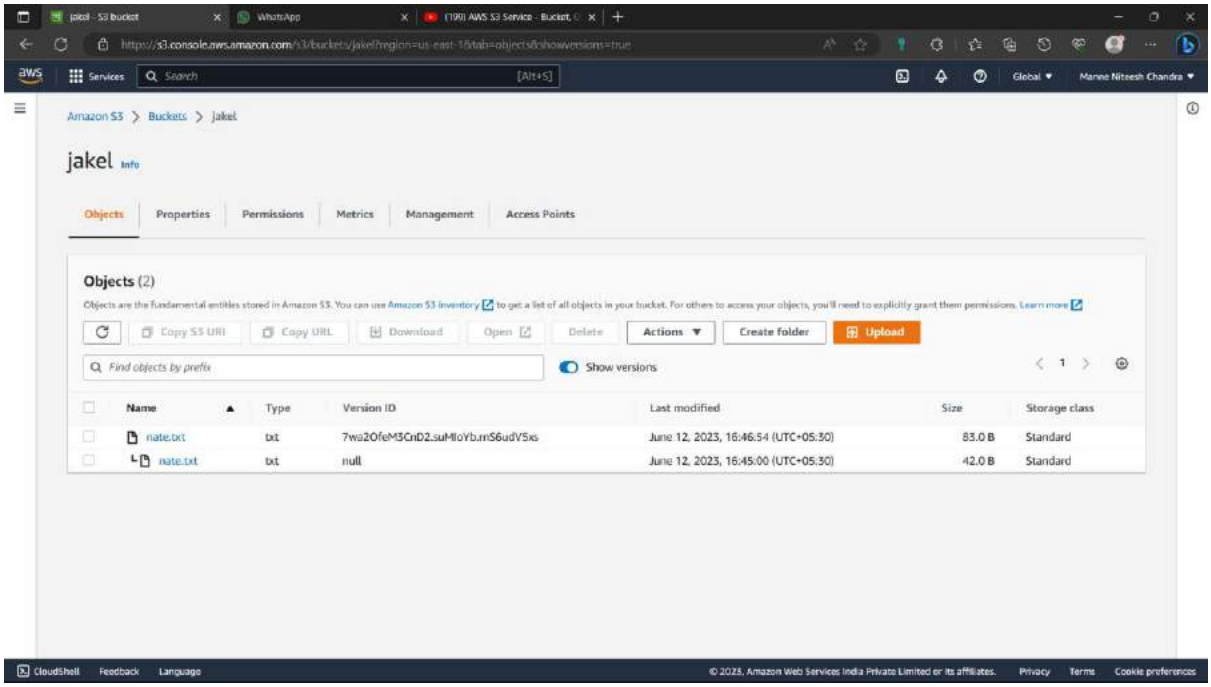
126)By deleting the marker file we can restore both the files. Select the marker file and click on delete option.



127)Type “permanently delete” and click on “delete object” option.

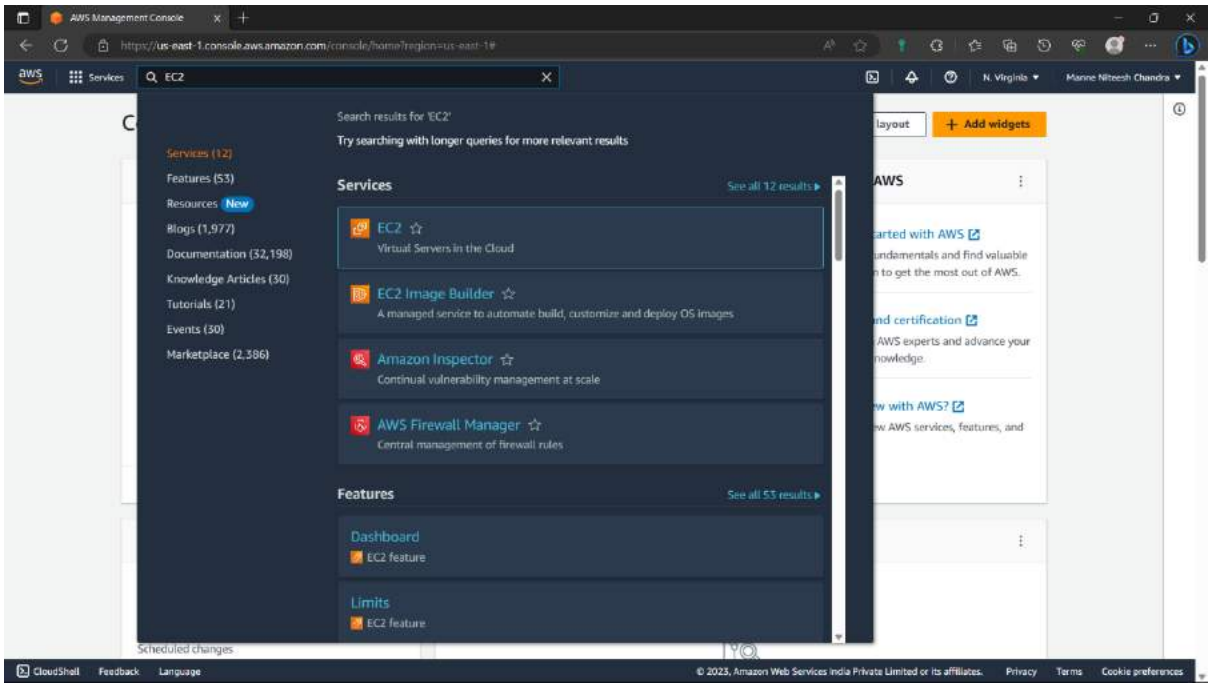


128)Now can see that both the deleted files are restored after deleting the marked file.

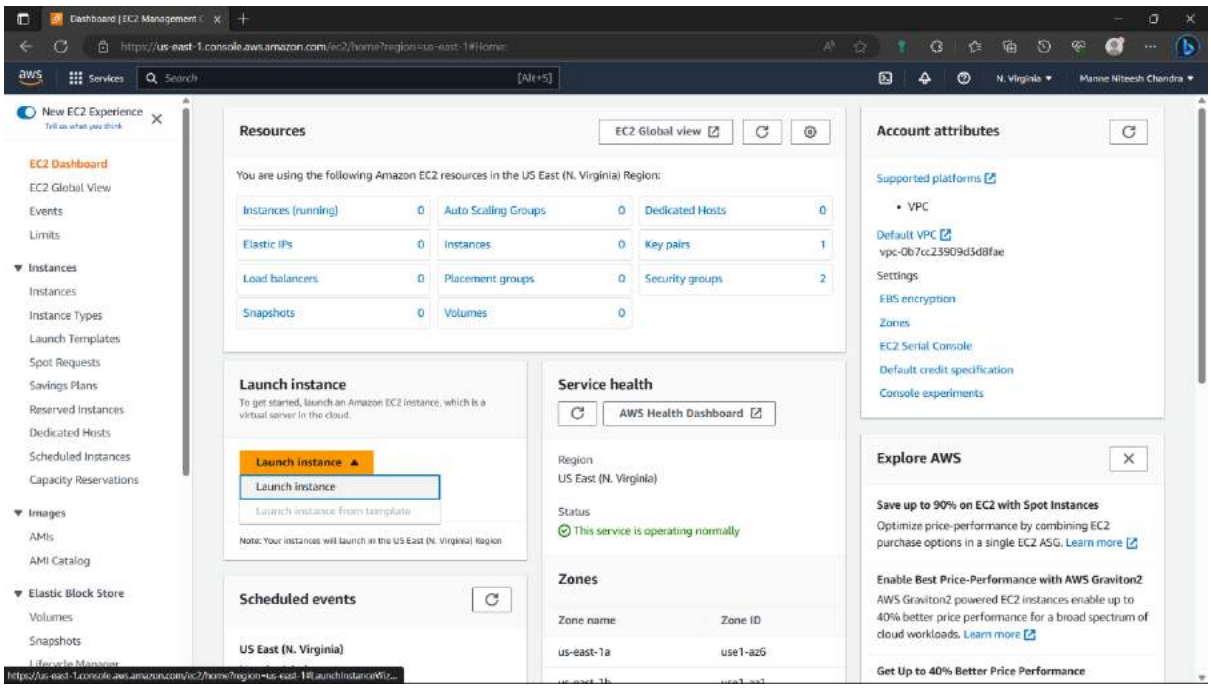


EC2 Instance:

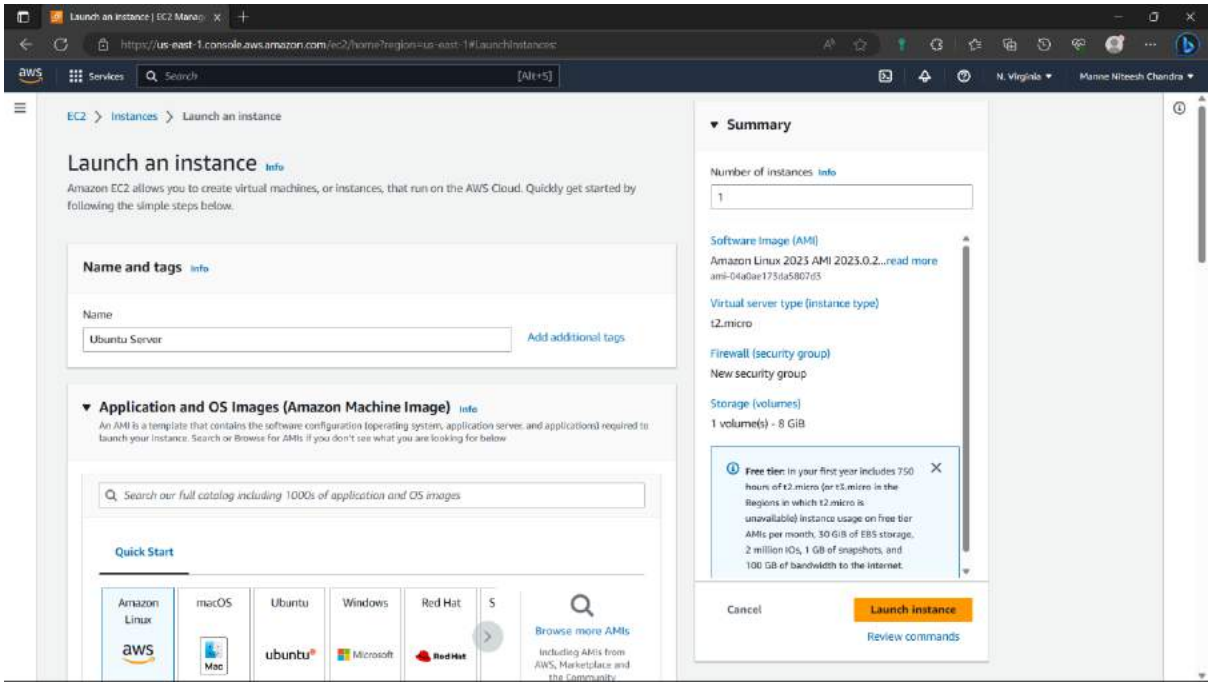
129)Type “EC2” in the search bar and select the “EC2” option below.



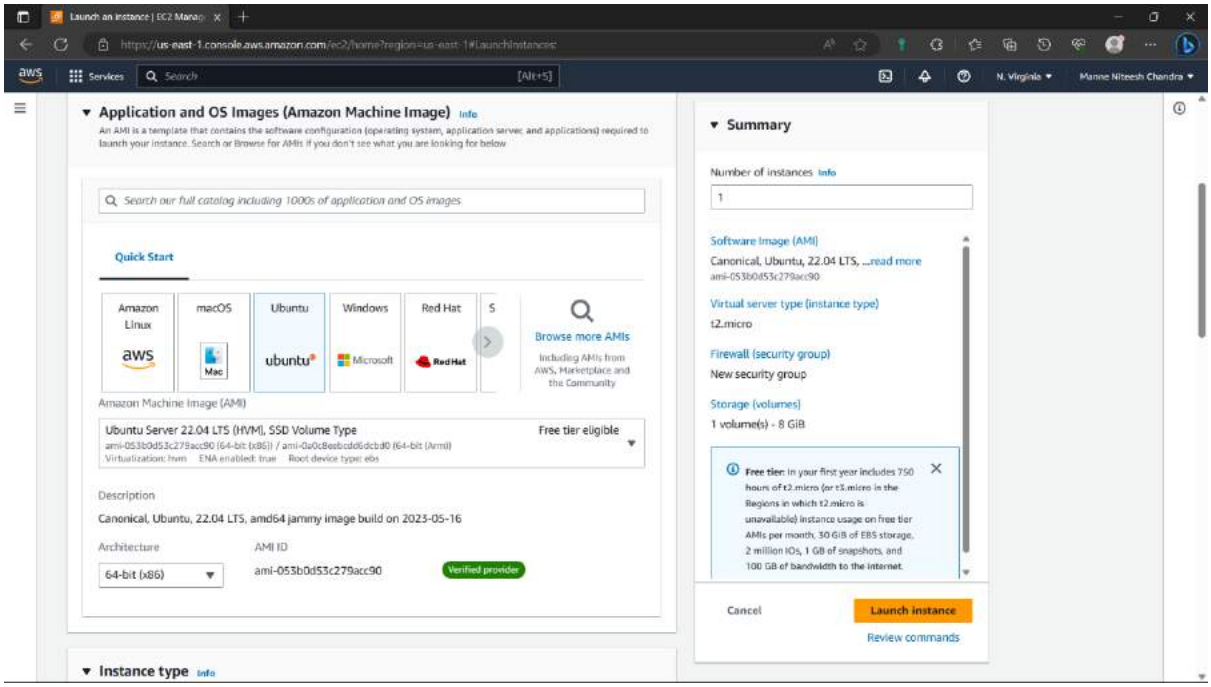
130)Click on “Launch instance” option to create an EC2 instance.



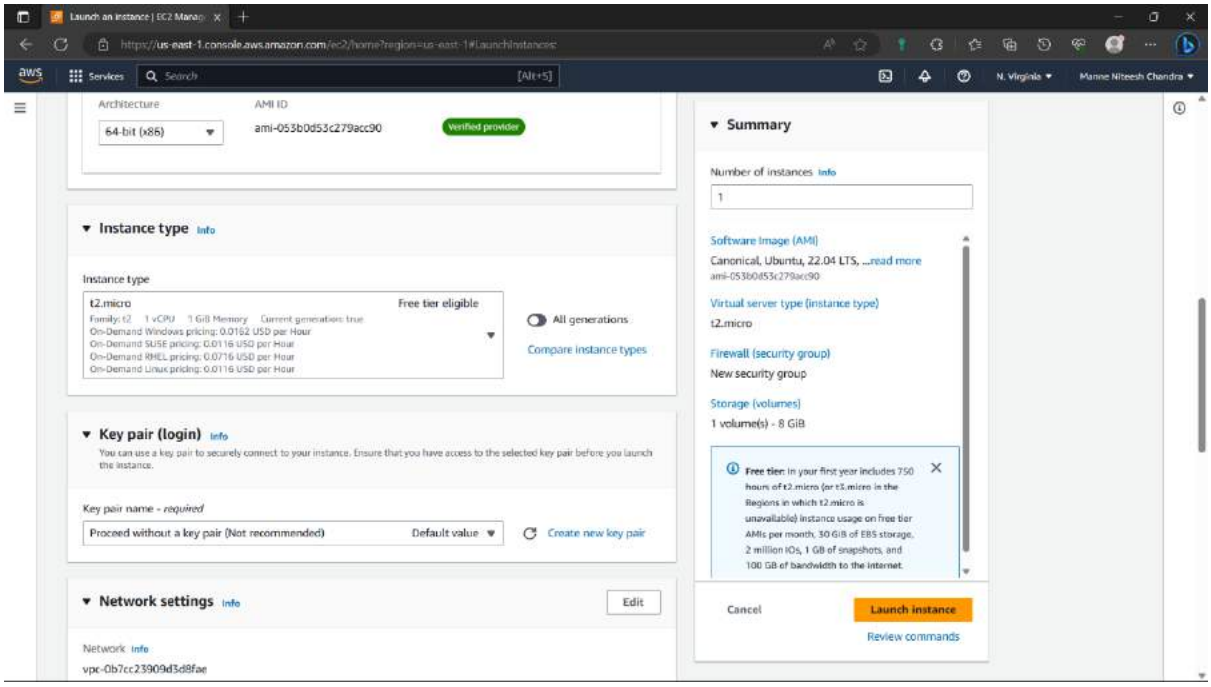
131)Give a name to the instance.



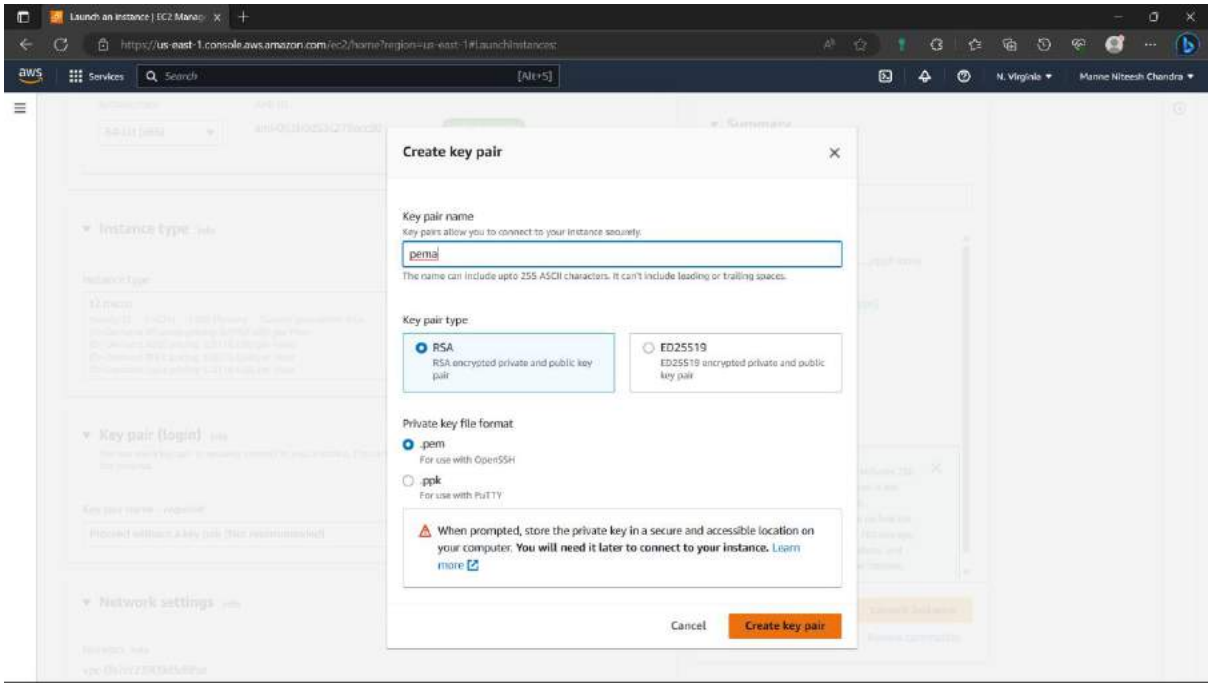
132)Select the AMI- Ubuntu OS, Description(default), Architecture(default), Am ID(default).



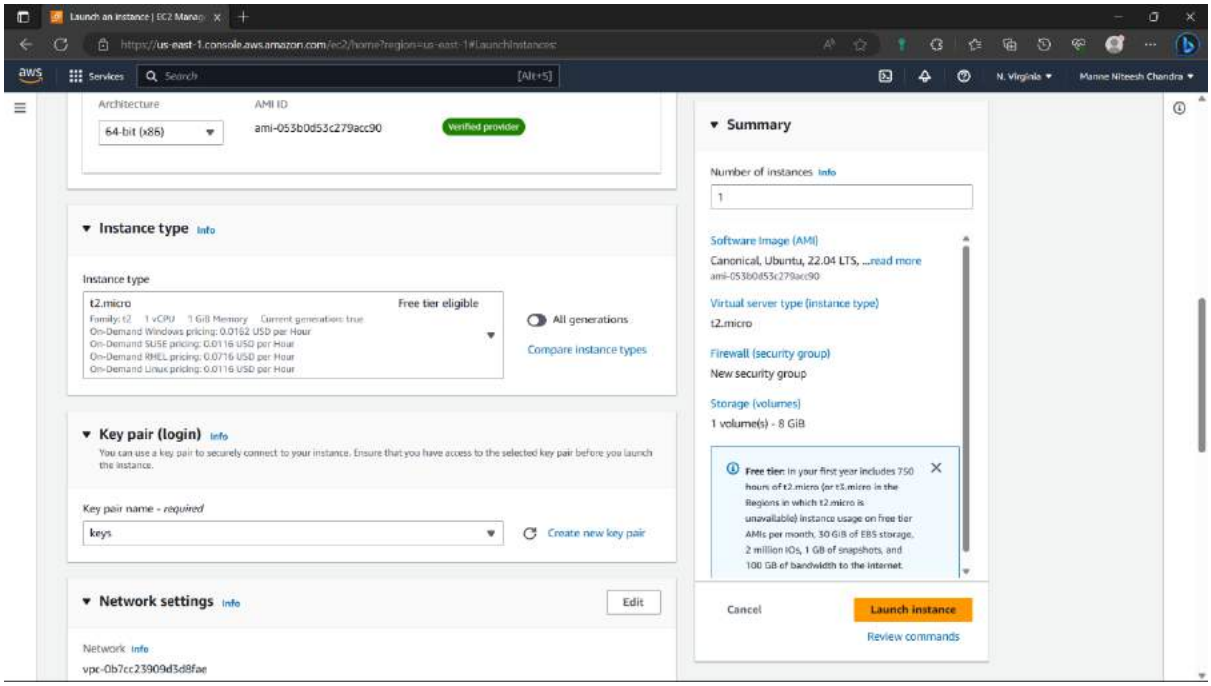
133)Select the instance type-t2.micro.



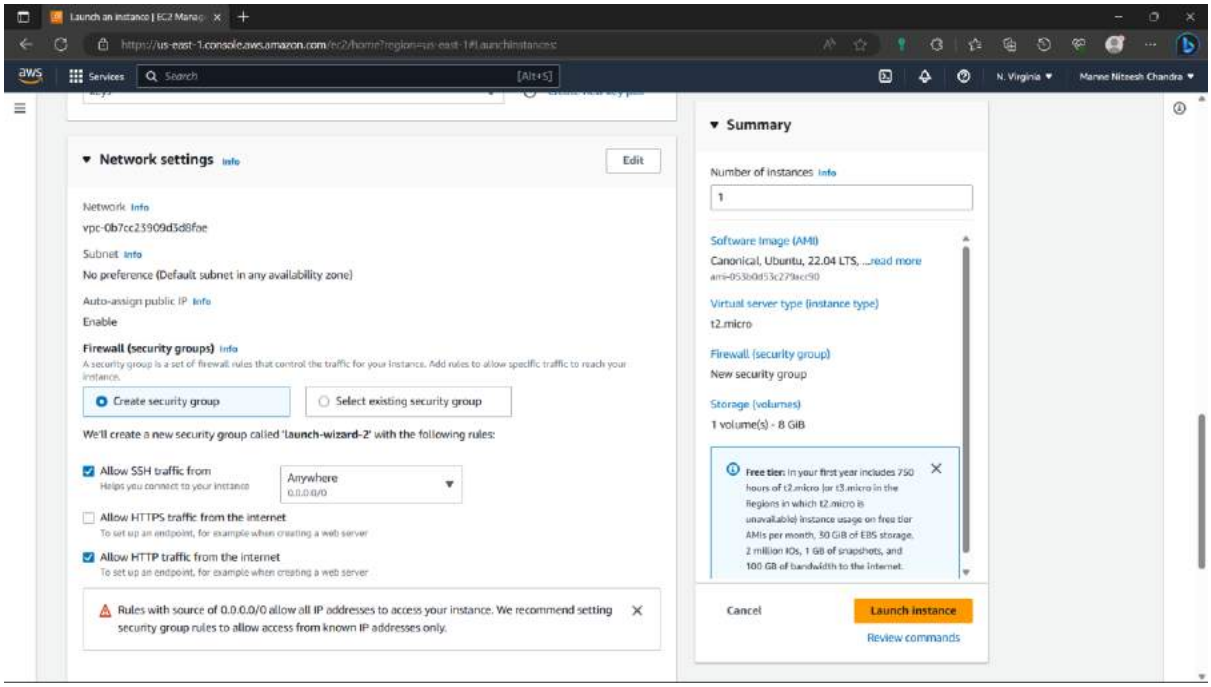
134)If you have the key pair select the key pair you have. Else click on “Create a new key pair” option. Give a name pof your choice and click on “create key pair”.



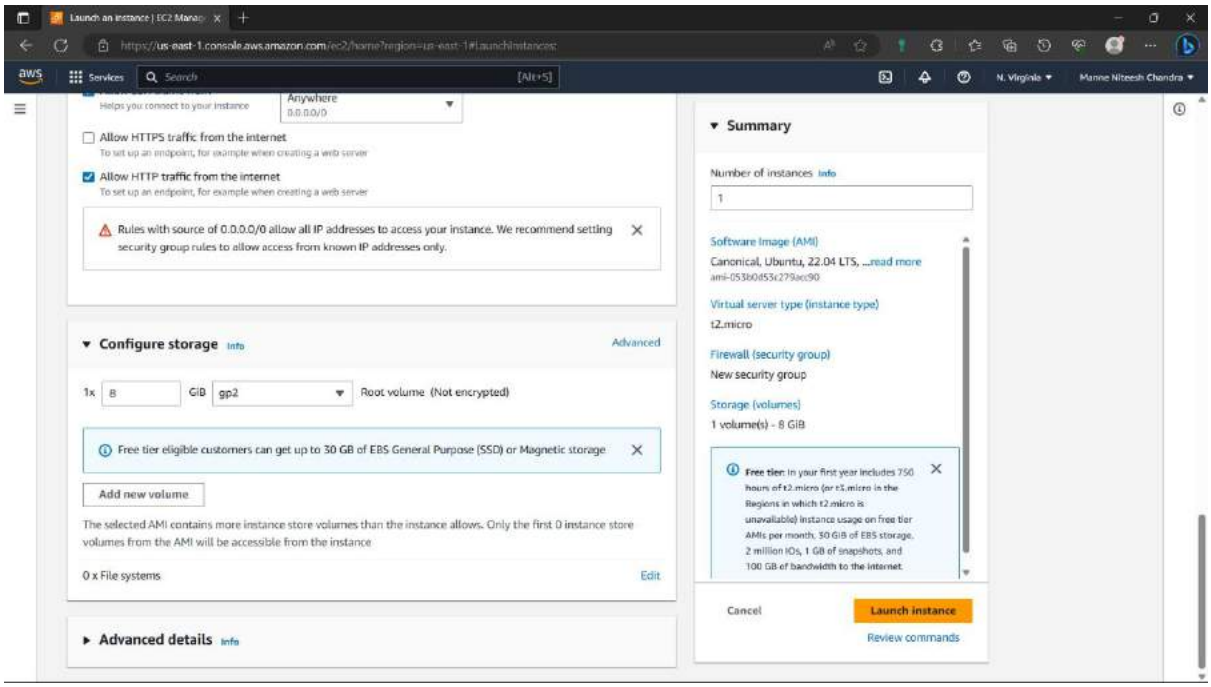
135)Now a copy of key pair is downloaded to the local system. And select the key pair you created in the system.



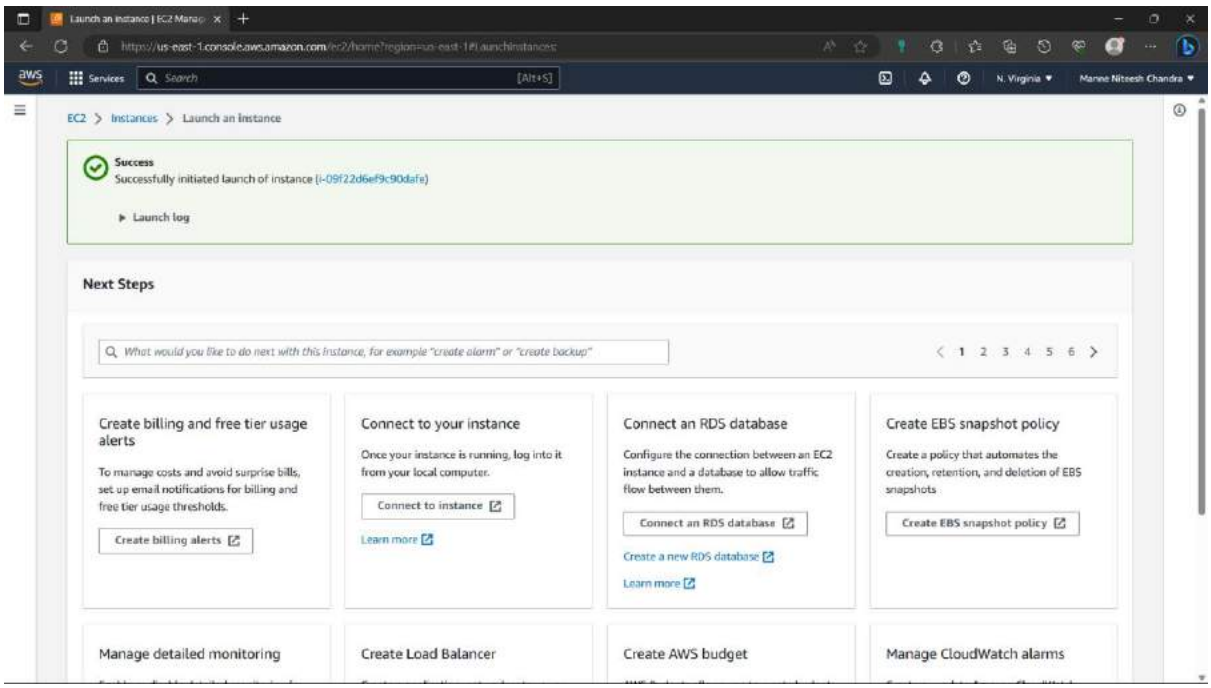
136)Enable the “HTTP Traffic” option in the network settings. So that it can be accessed through the URL.



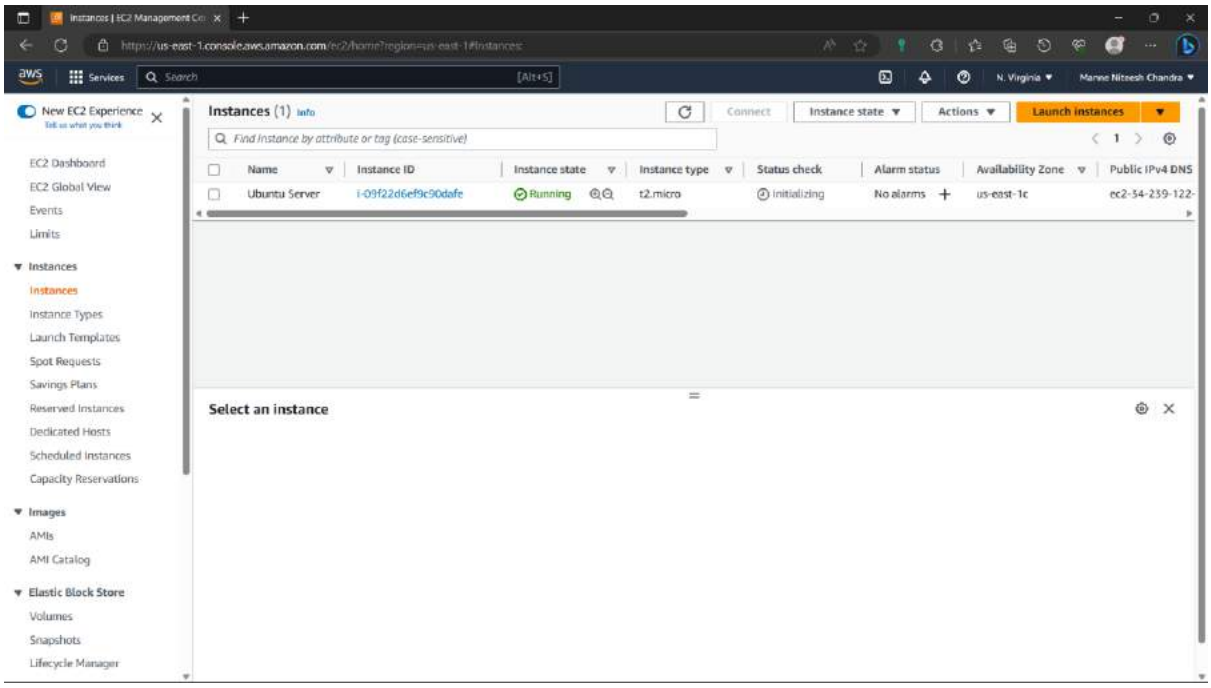
137)Review all the details you given in the summary and click on Launch instance” option to create the instance.



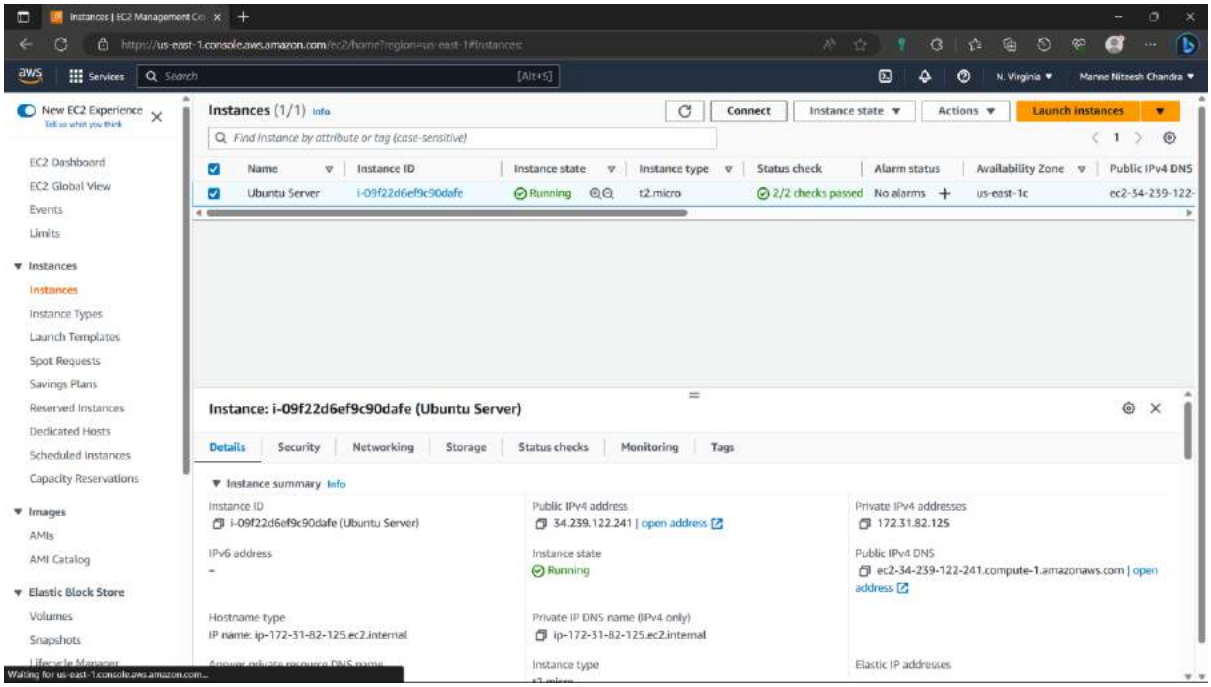
138)Now the instance is created.



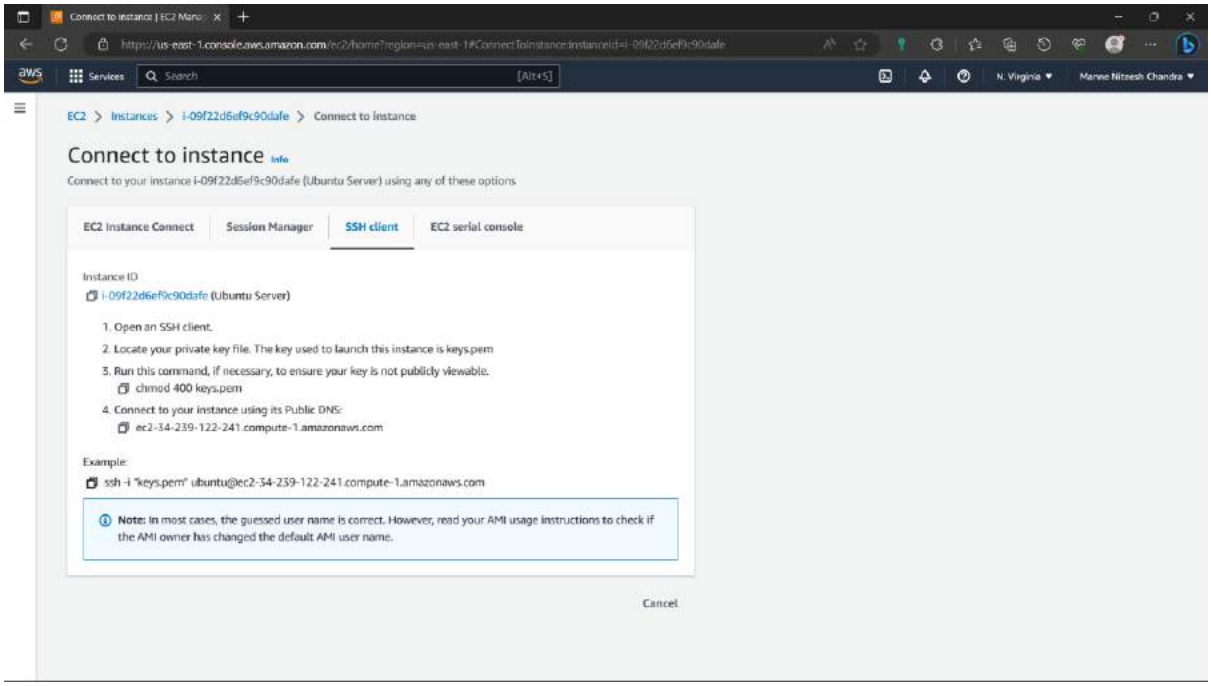
139)Now go to the instances. We can see the created instance. Wait till the Instance state- Running, Status type -2/2 checks passed.



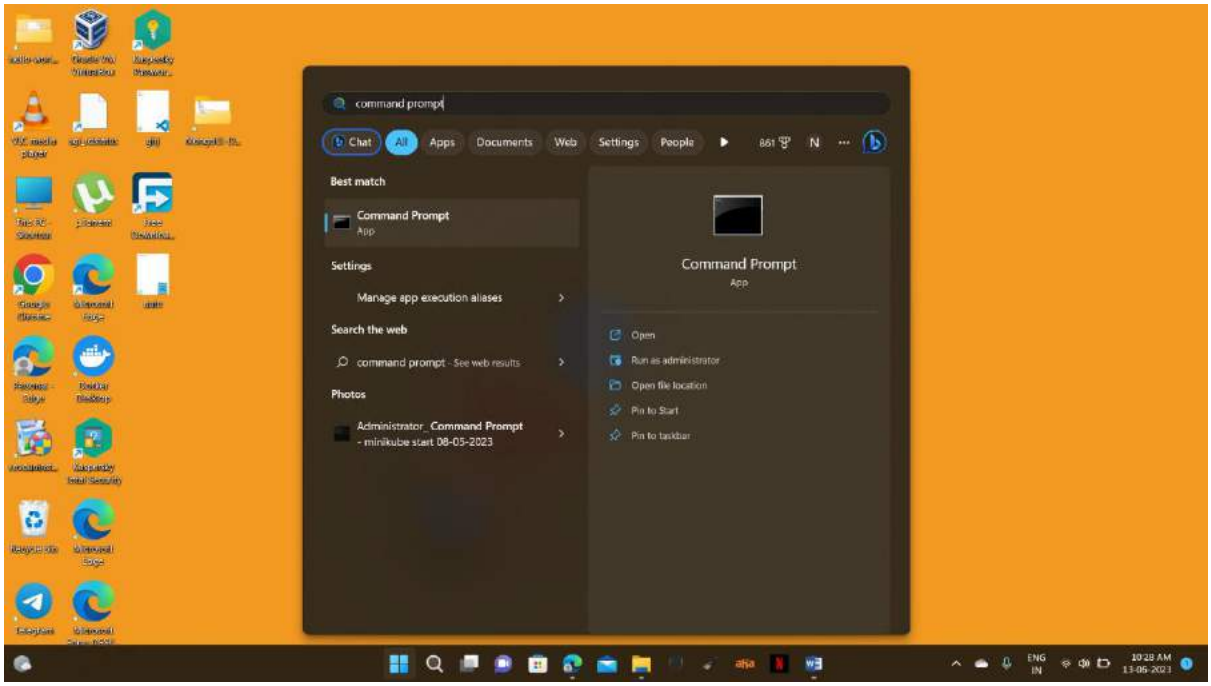
140)Refresh the instance, now the instance is ready. Select the created instance click on “connect”.



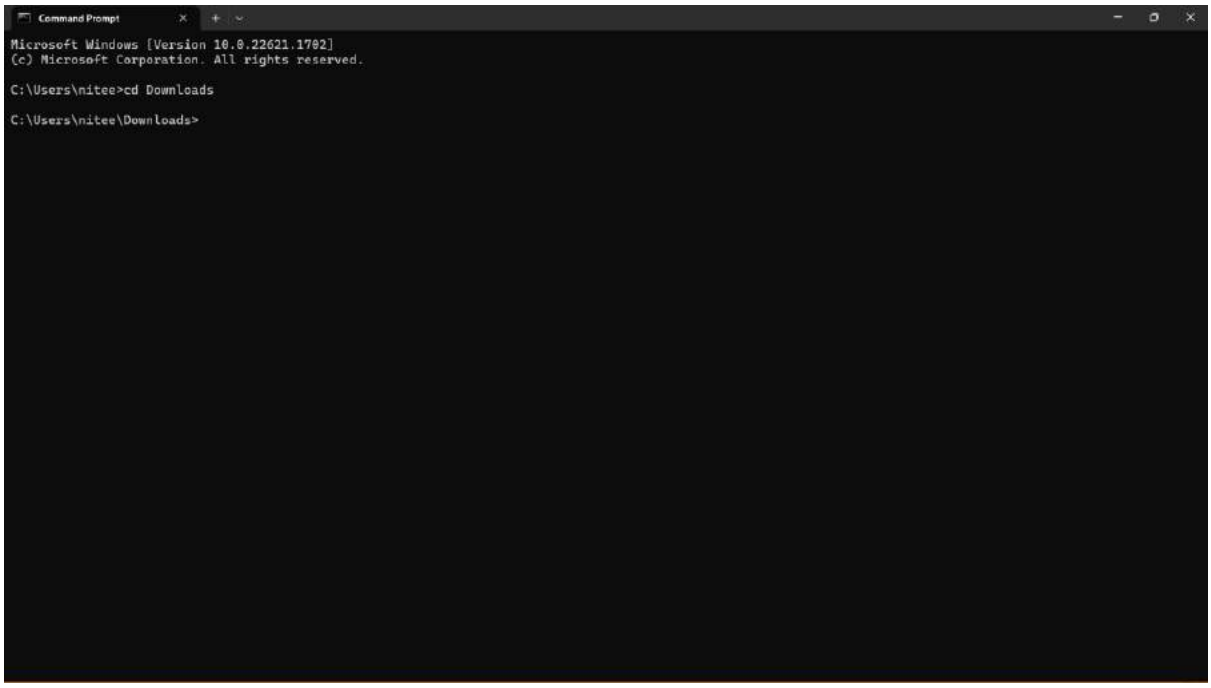
141)We are going to access the instance through local machine. Browse to “SSH client” option and copy the command below the “Example”.



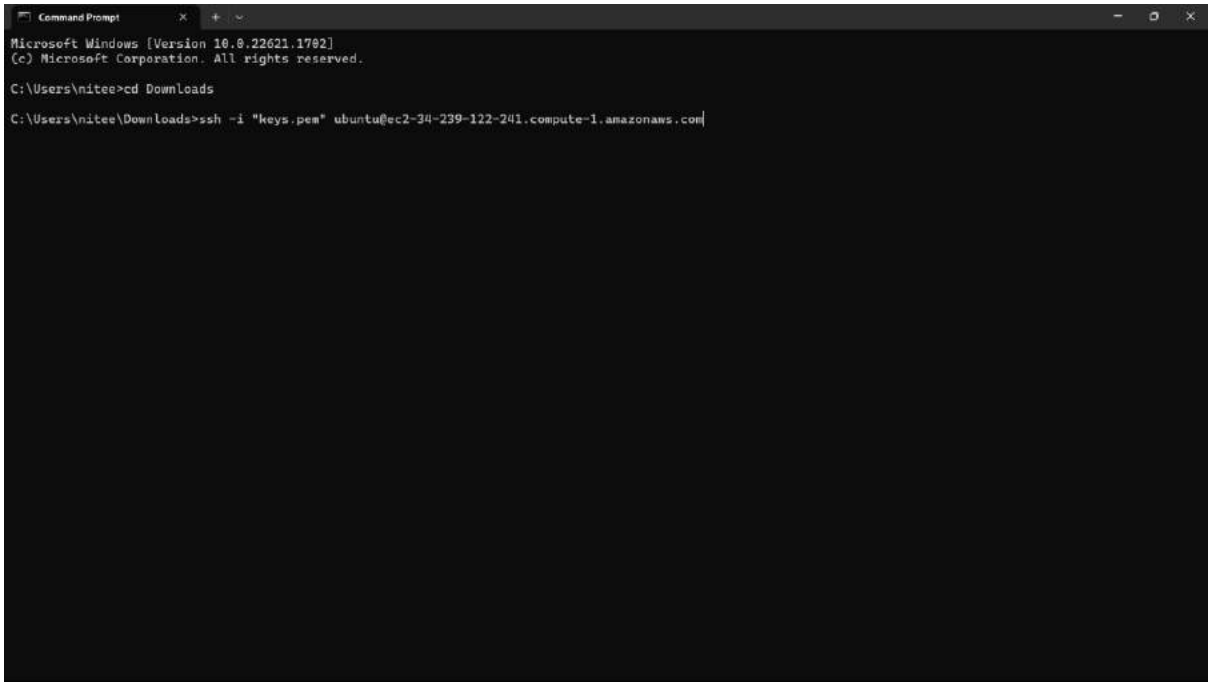
142)Open the command prompt in your local machine.



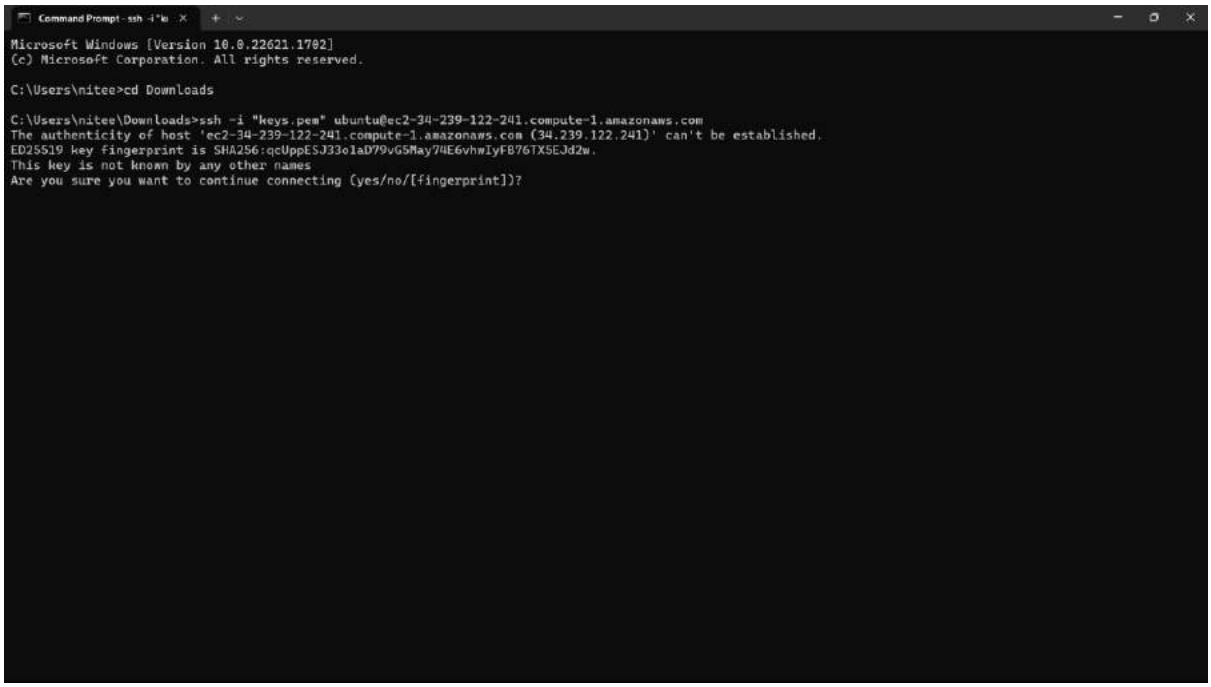
143)Type the command – “cd path” path-Where the key pair is located in my local system the key pair is in downloads so my command- “cd downloads”.



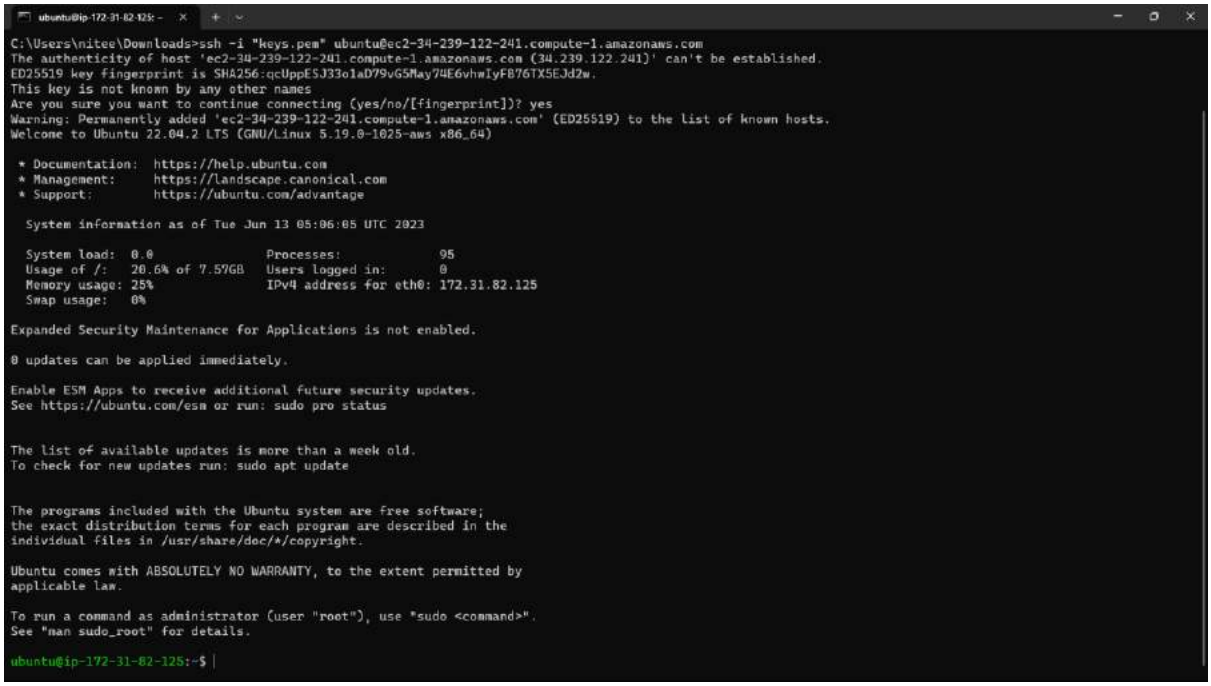
144)Now paste the command you copied in “step-141”(because it is the command which is used to connect with the instance locally and key pair is the middleman between the ec2 at AWS and ec2 in the command prompt) and click on enter.



145)type “yes” click enter button in the keyboard.

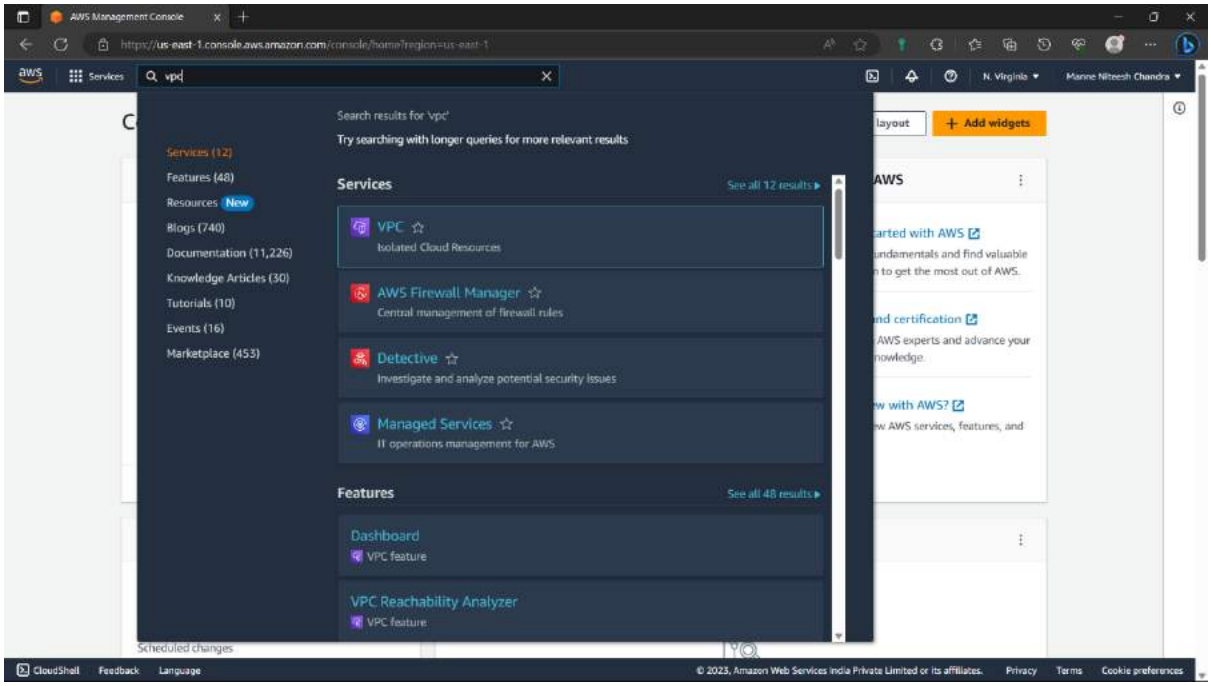


146)Now the instance is created in your local system.

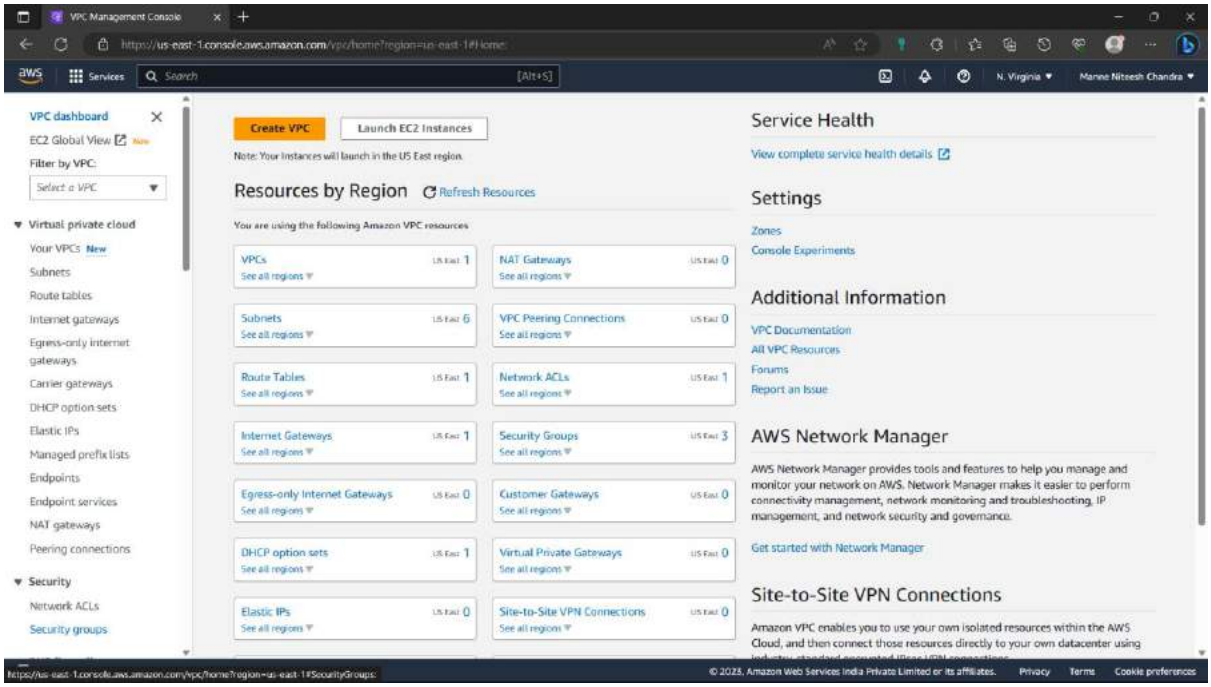


Security Group:

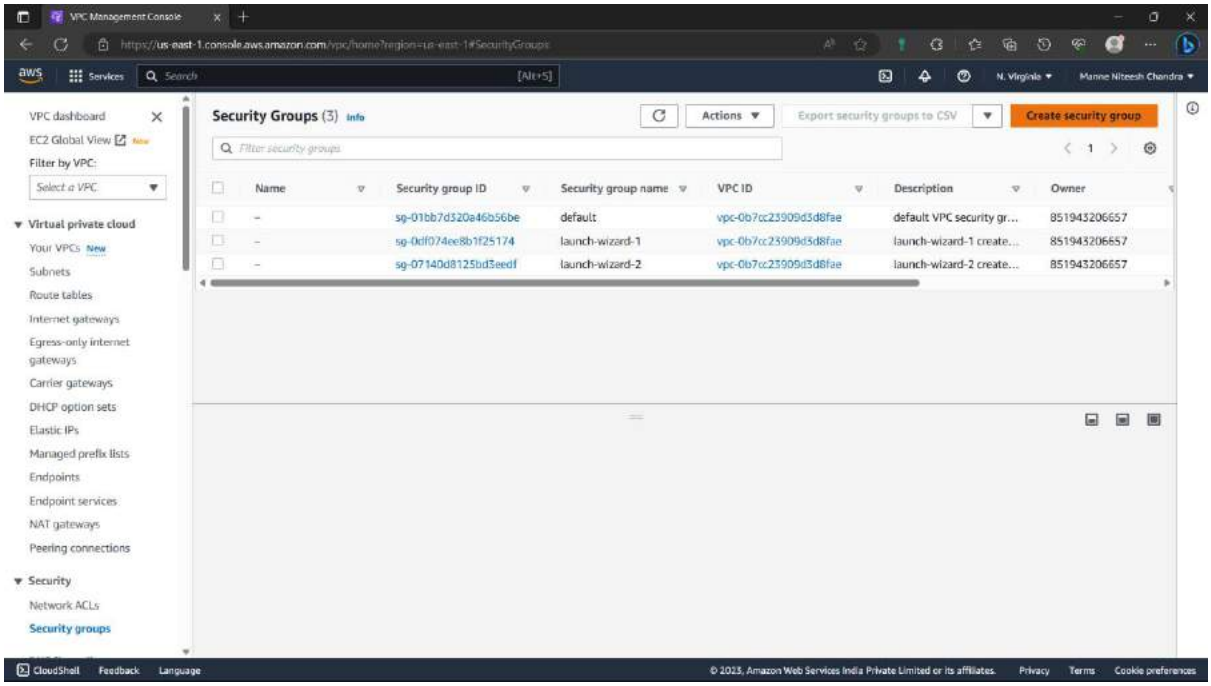
147)Go to the search bar in AWS and type “VPC”, Click the “VPC” option.



148)In the navigation panel below on the left side in “security” there is “security group” option, click on the “security group” option.



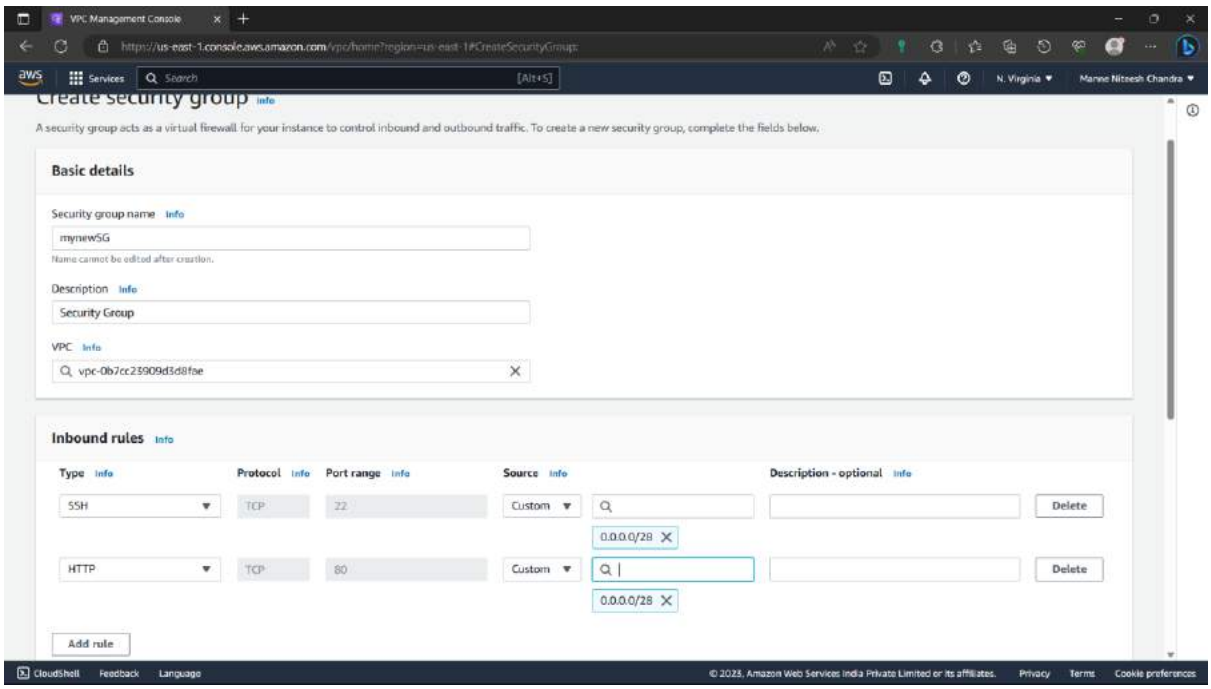
149)Click on “create security group” option at top right corner in yellow colour.



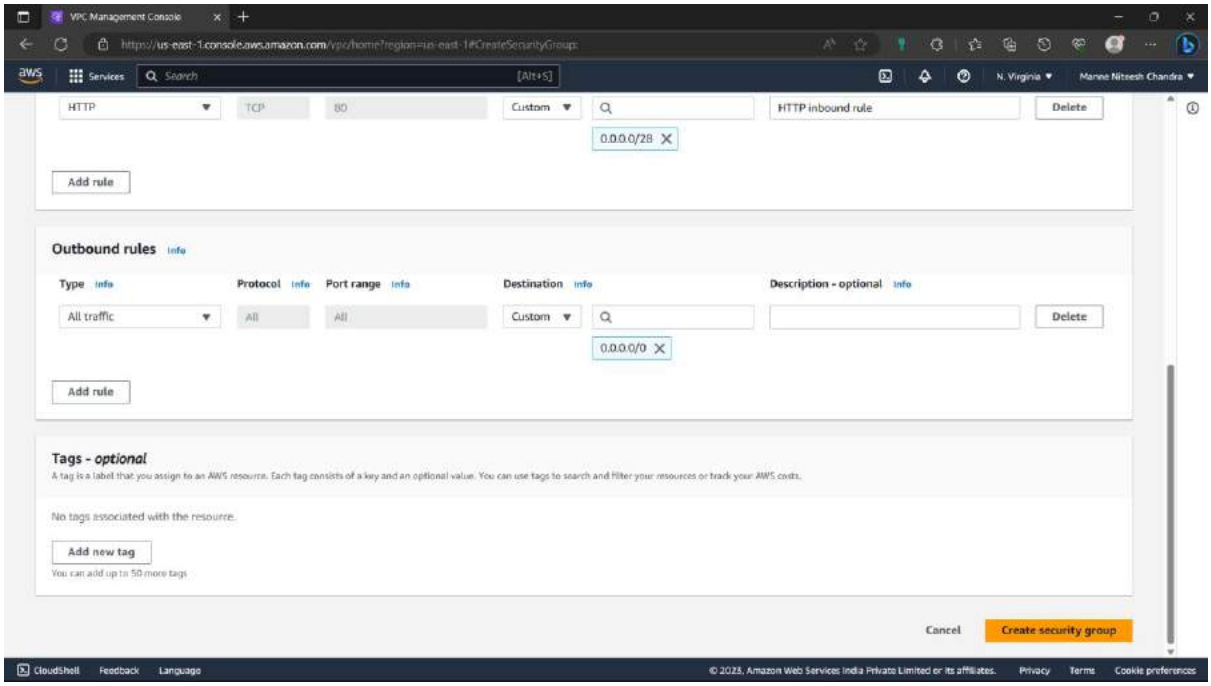
150)Give name to the security group (you can identify it easily among all the security groups) let the VPC be default. Click add rule in inbound rule add 2 rules

i)Type- SSH, port-20, IP range-0.0.0.0/28, Description(your choice).

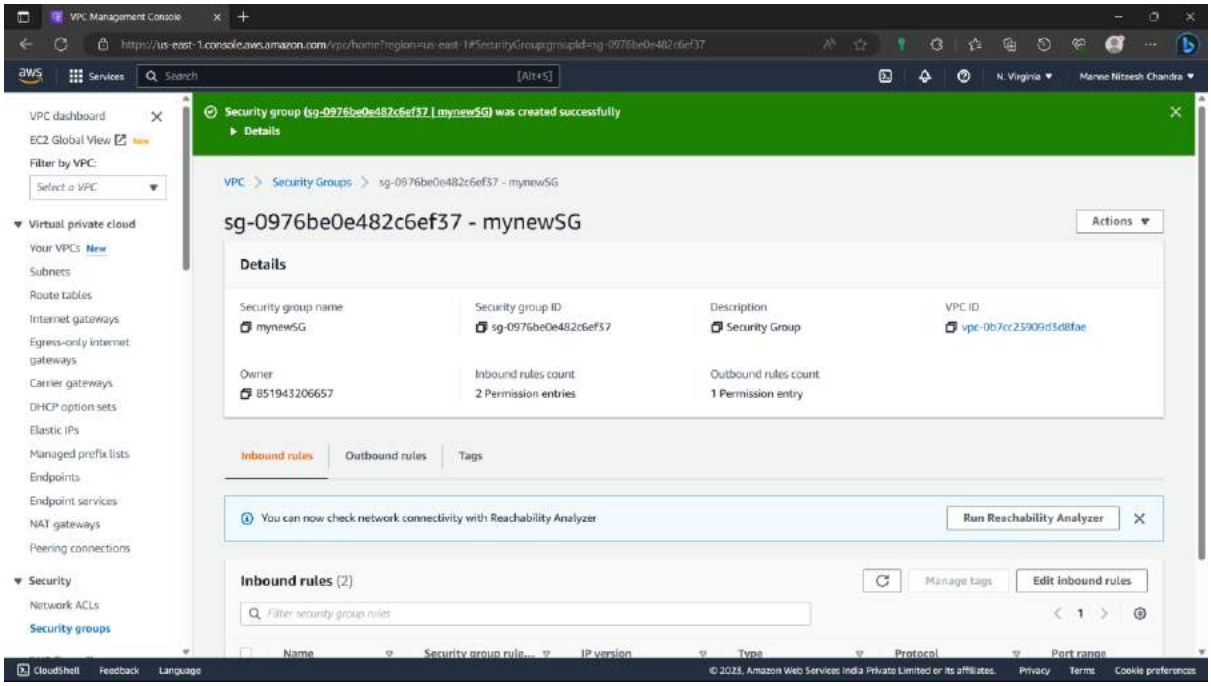
ii)Type-HTTP, port-80, IP range-0.0.0.0/28, Description(your choice).



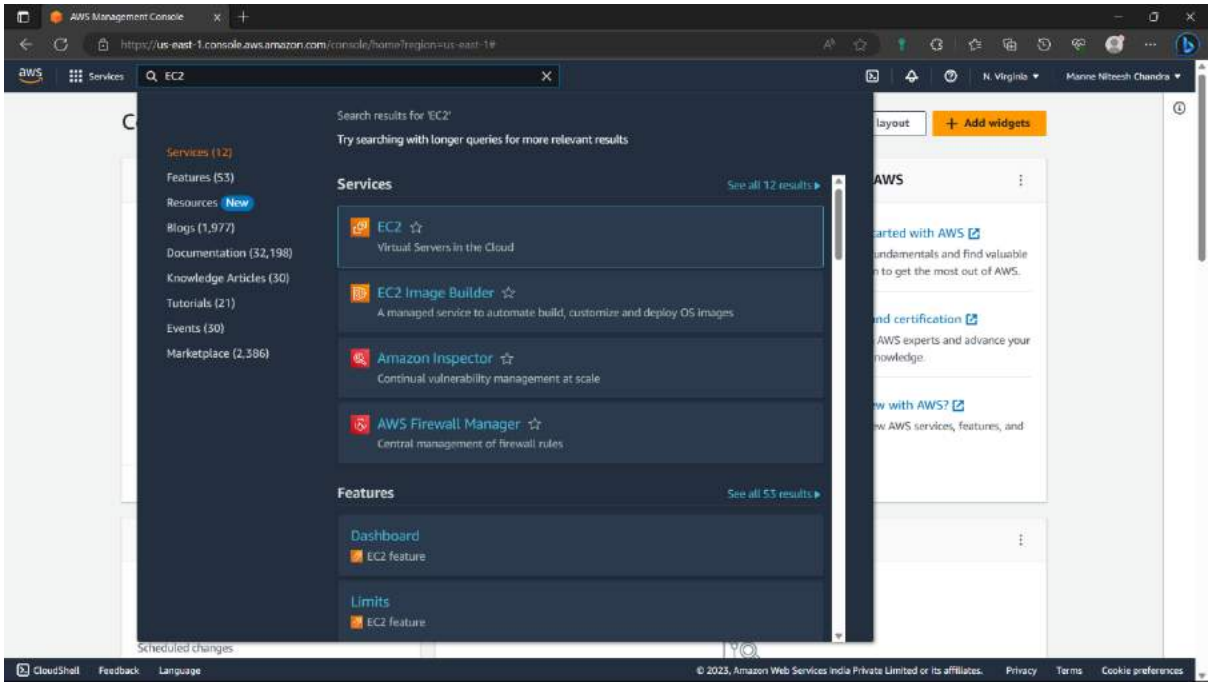
151)Click on “create security group”.



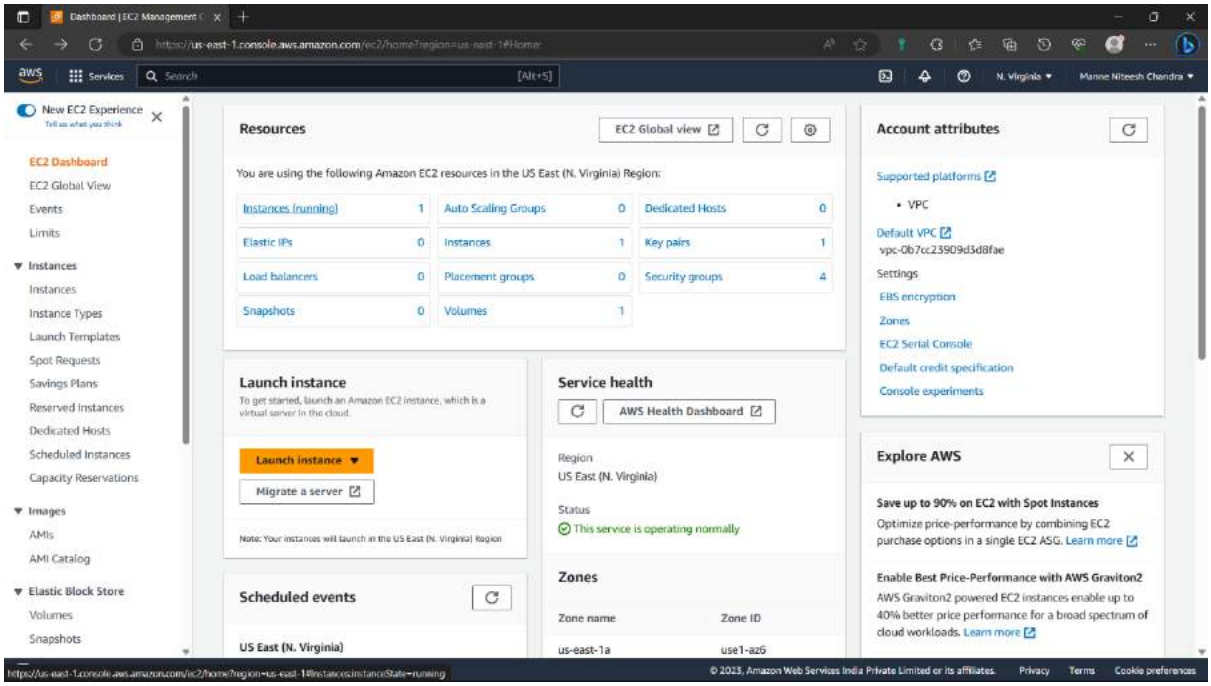
152)The security group is created.



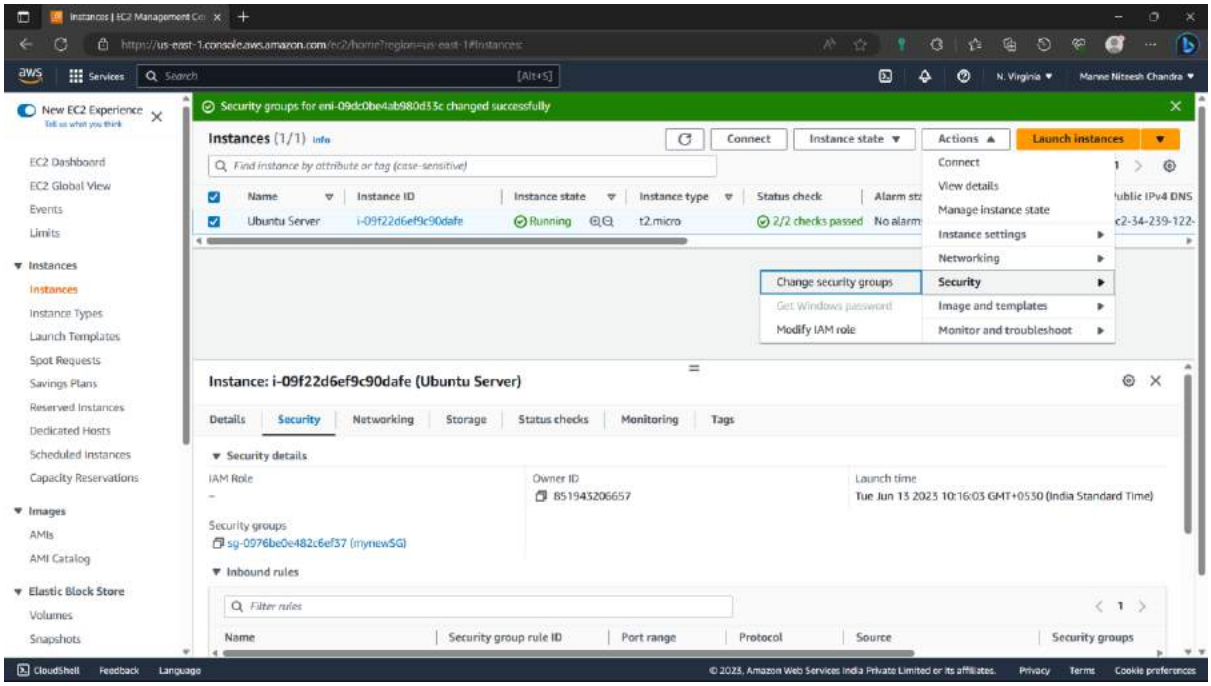
153)Go to search bar type “EC2” and select the “EC2” instance option.



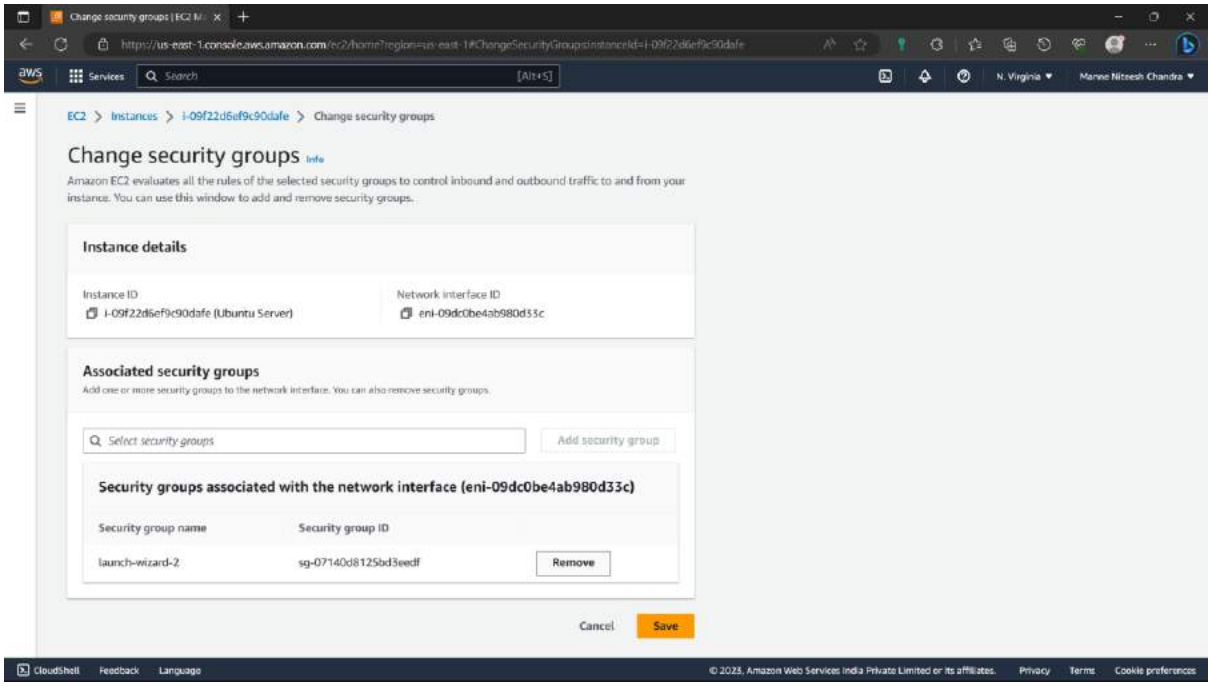
154)Click on “Instance(running)” option.



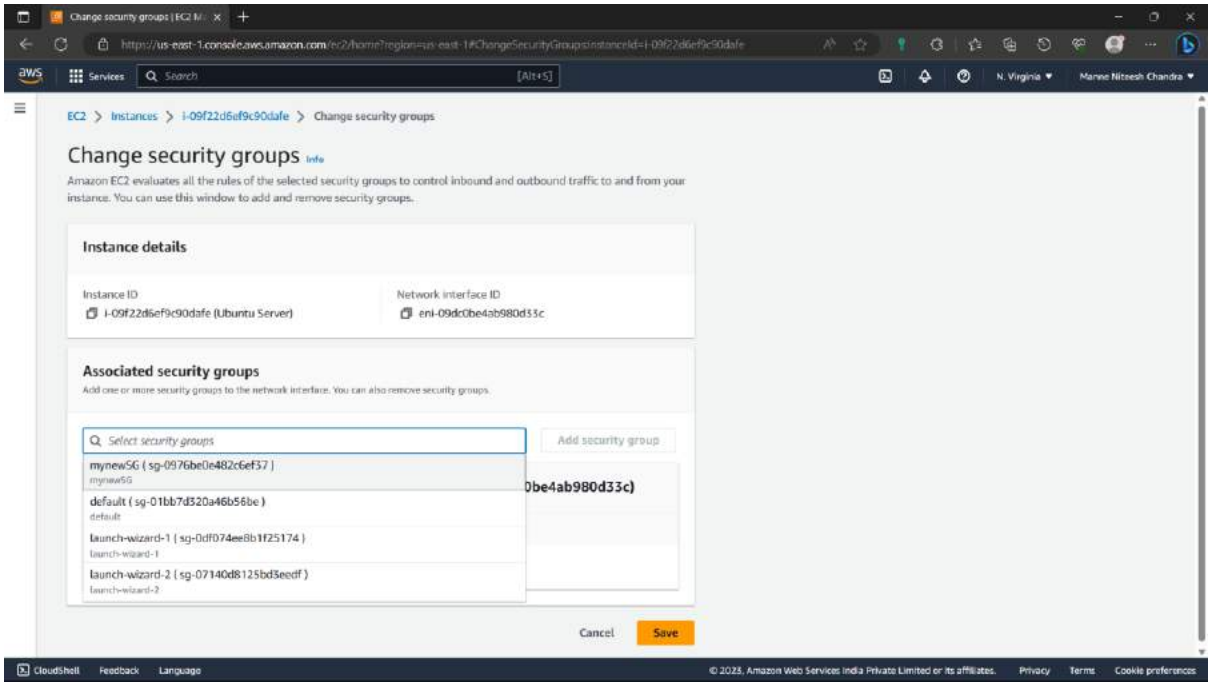
155)Select the instance you created in steps(131-138) and click “Actions”→”Security”→”change security group”.



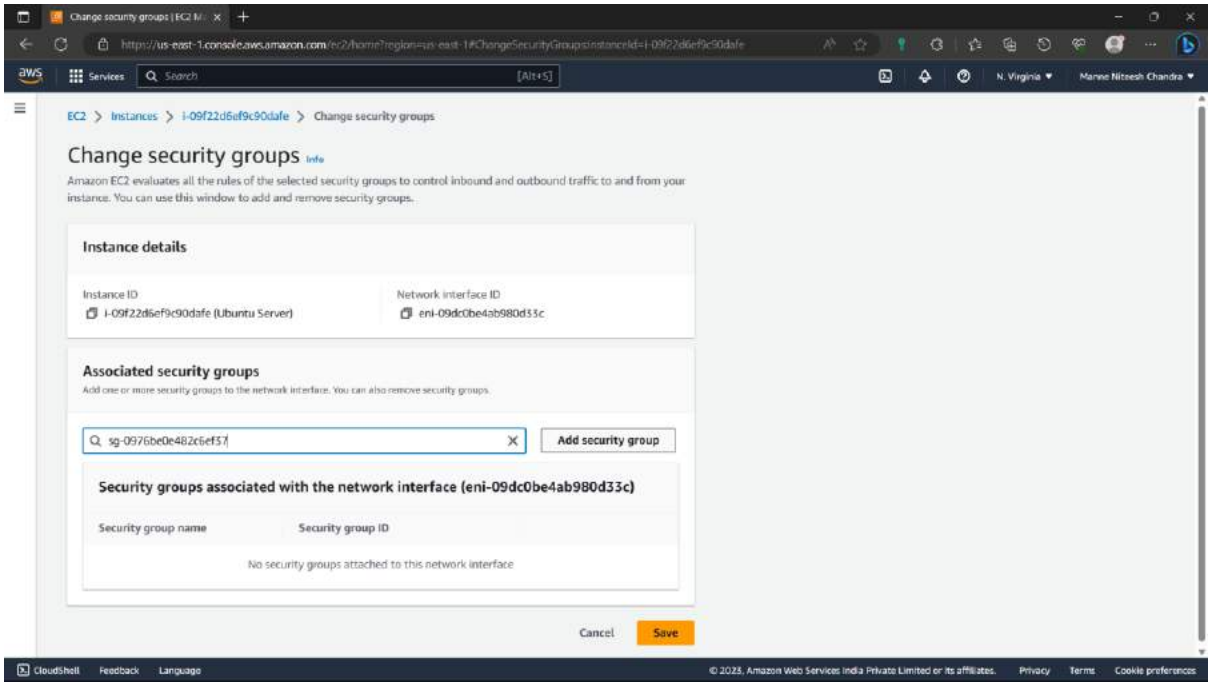
156)Remove the security group i.e., assigned default.



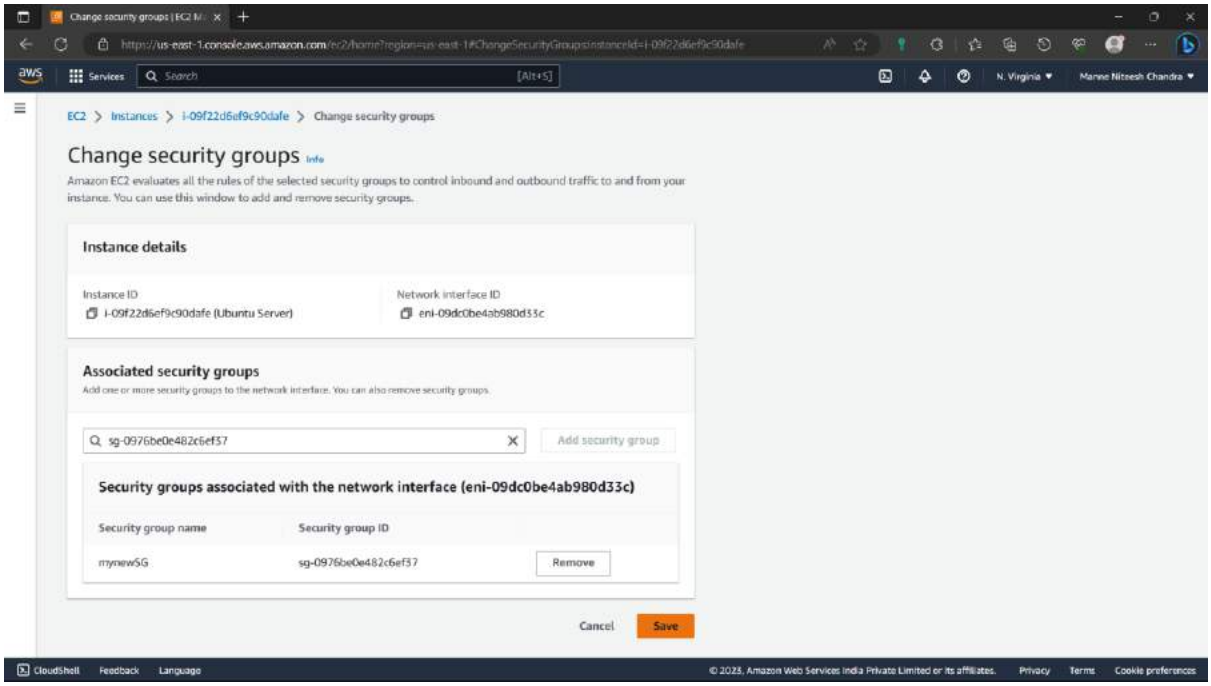
157)In search bar of “Associated security group” select the security group we created (Steps:149-152).



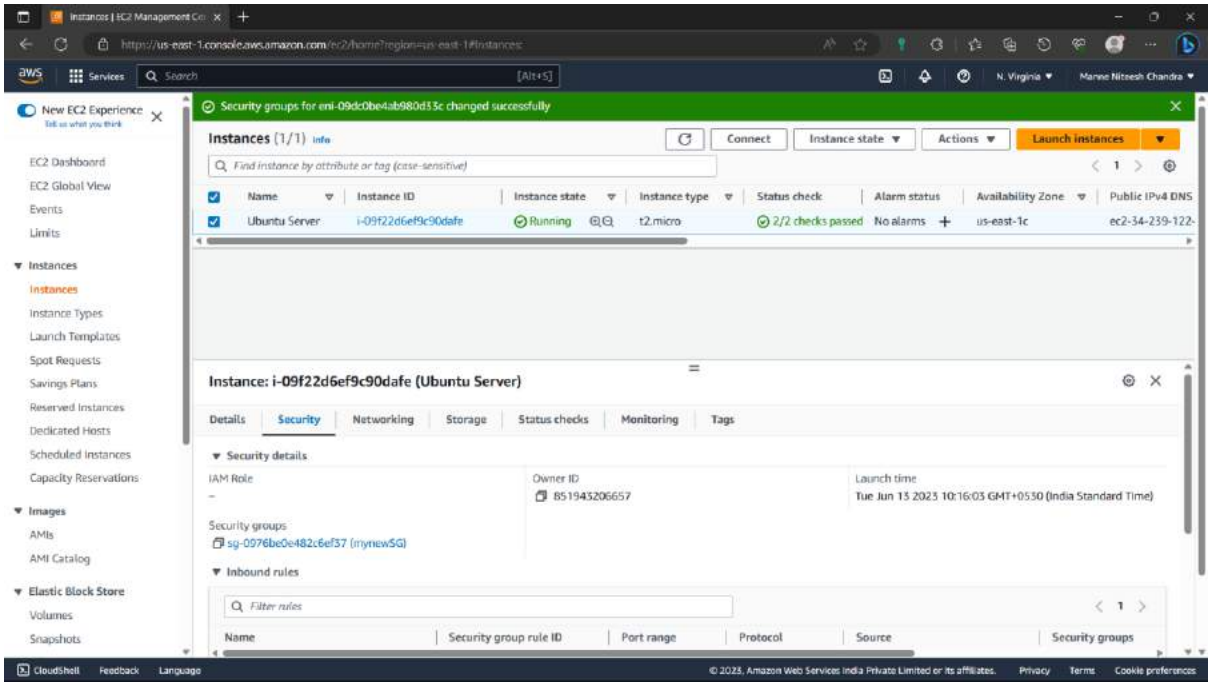
158)Click on “Add security group”.



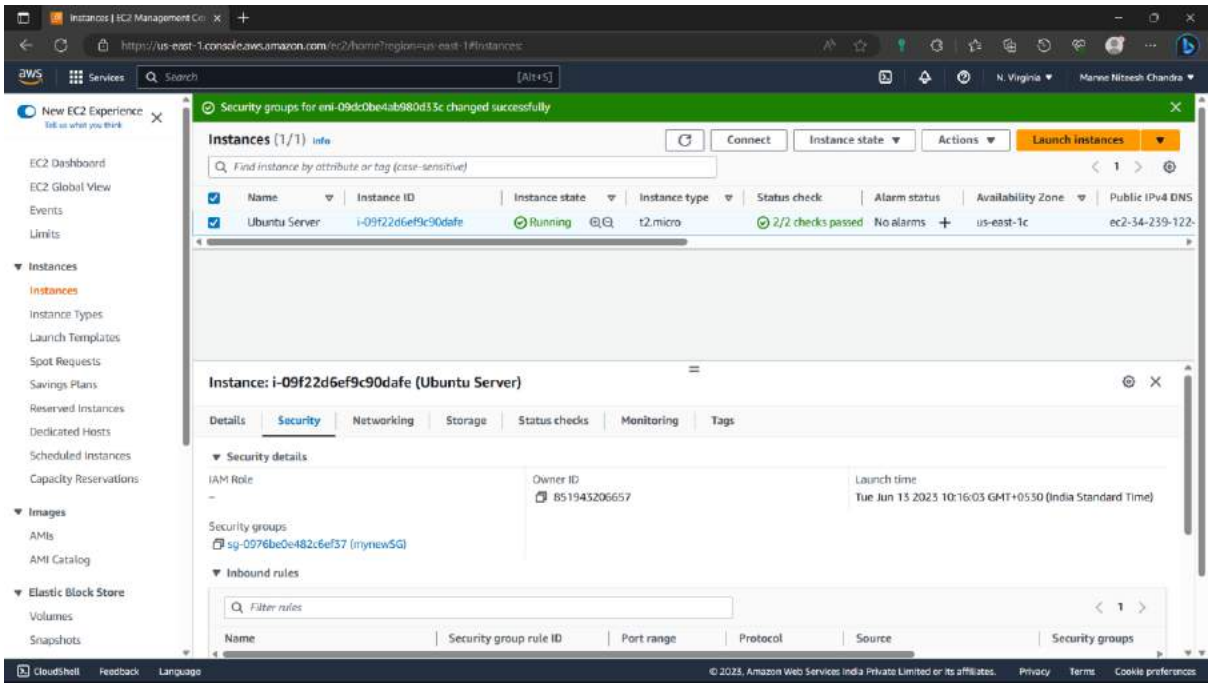
159)Click on “save” option.



160)Now the security group is added to the instance.

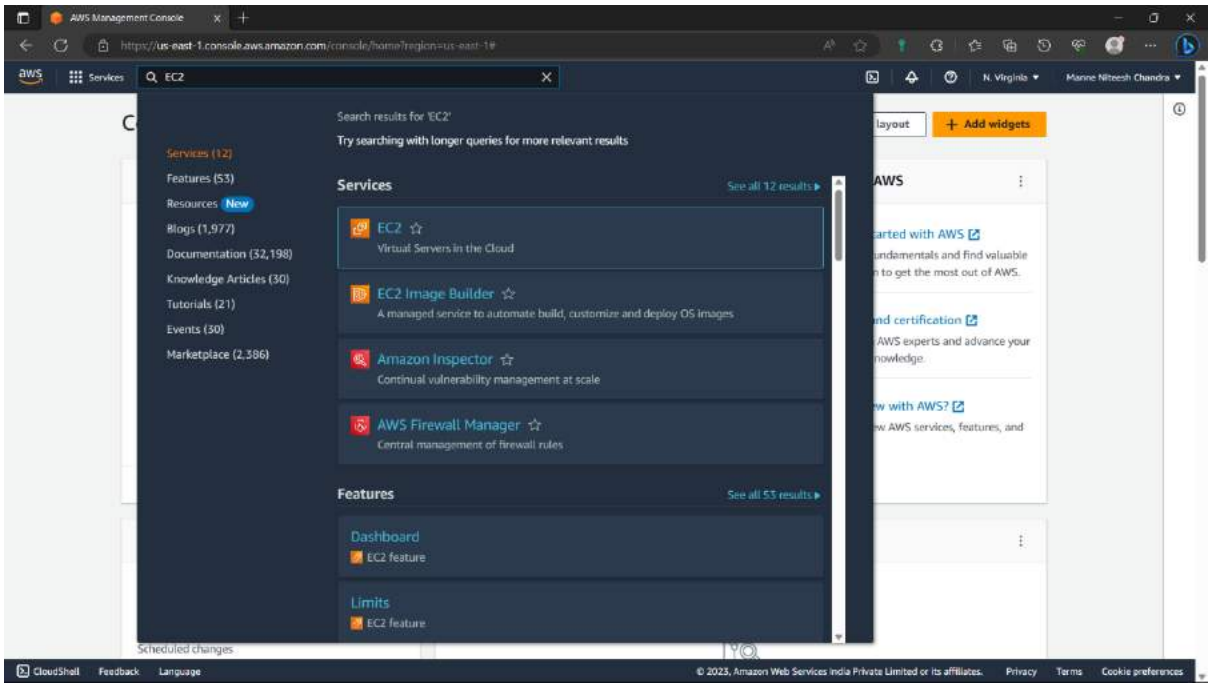


161)You can check it by selecting the security group assigned ec2 instance and navigate to the “security” option of the instance. In security we can see the assigned security group in “security group” option.

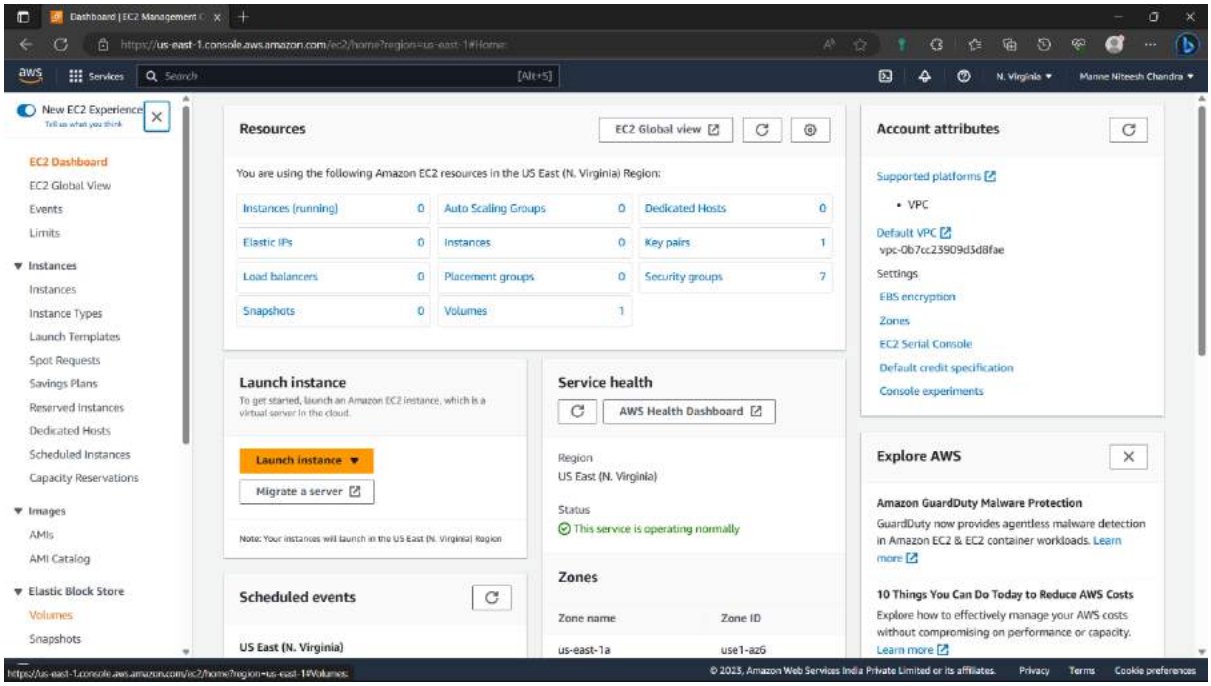


Volume and Snapshot:

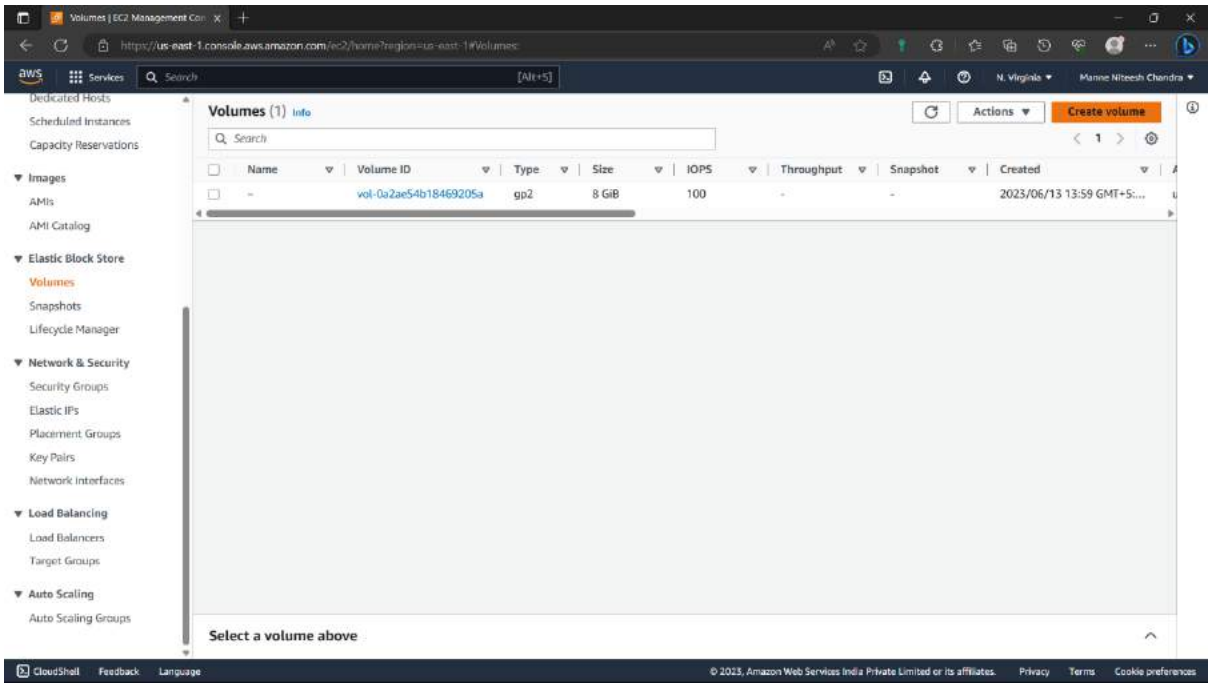
162)Type “EC2” in the search bar and select the ”EC2” option below.



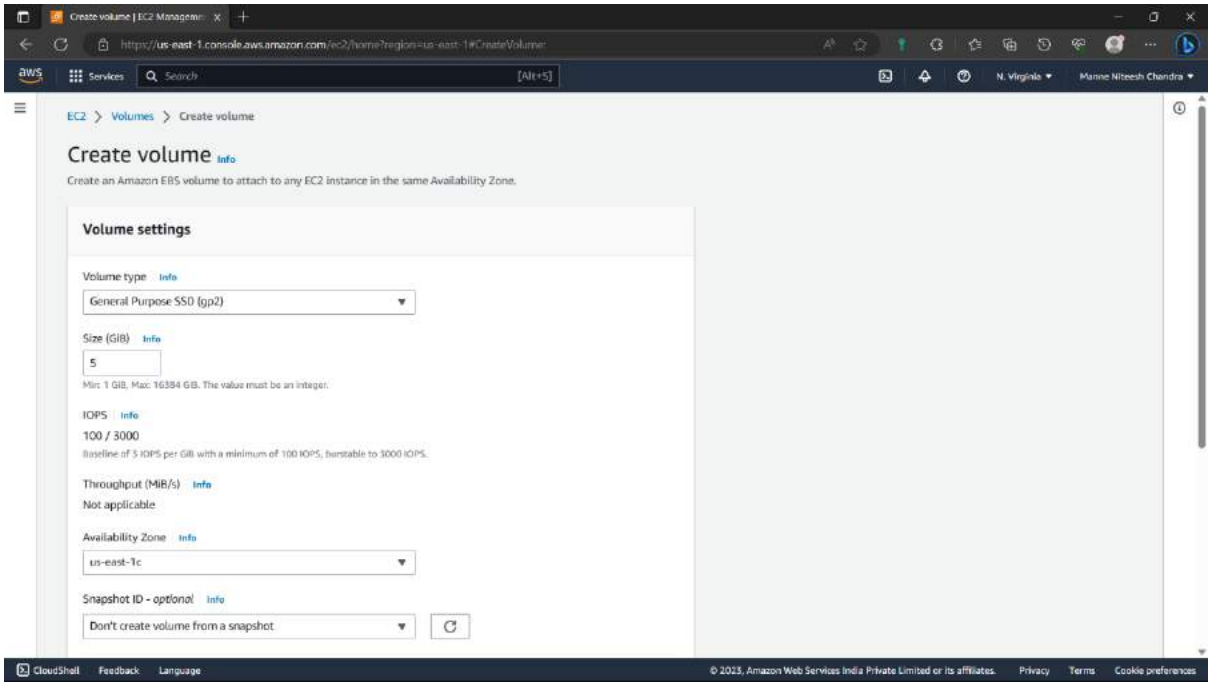
163)Select volumes in “Elastic block store” i.e., bottom of the navigation panel of the EC2 instance.



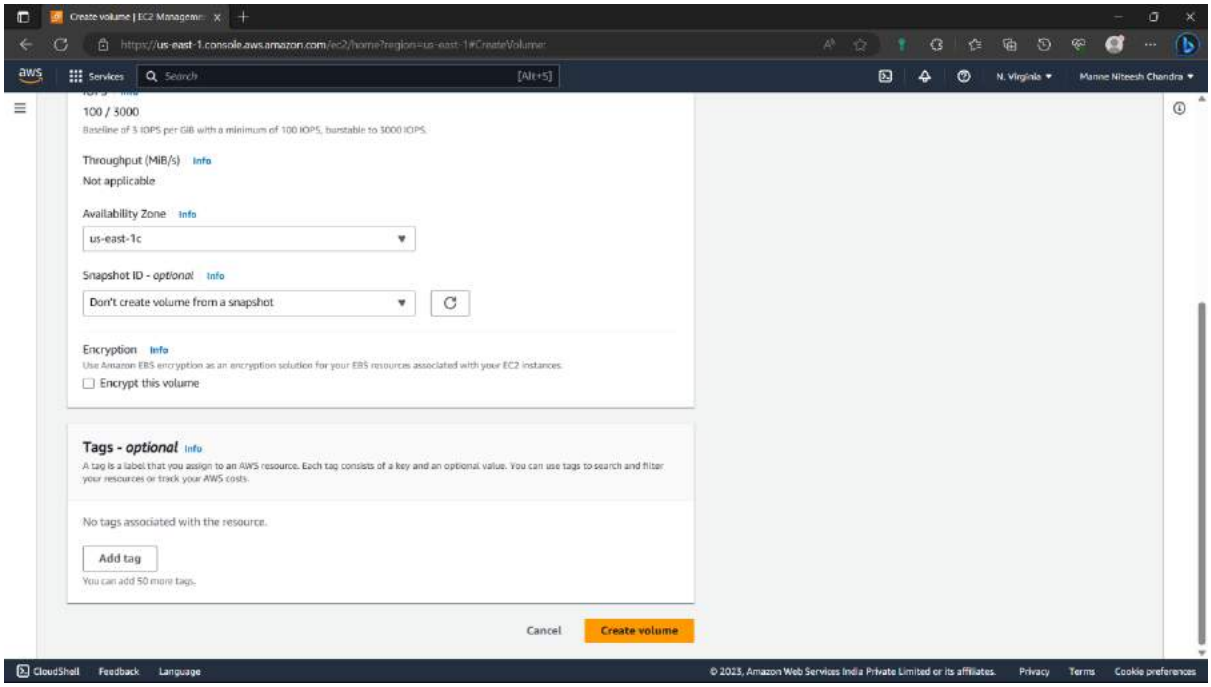
163)Click on “create volume”.



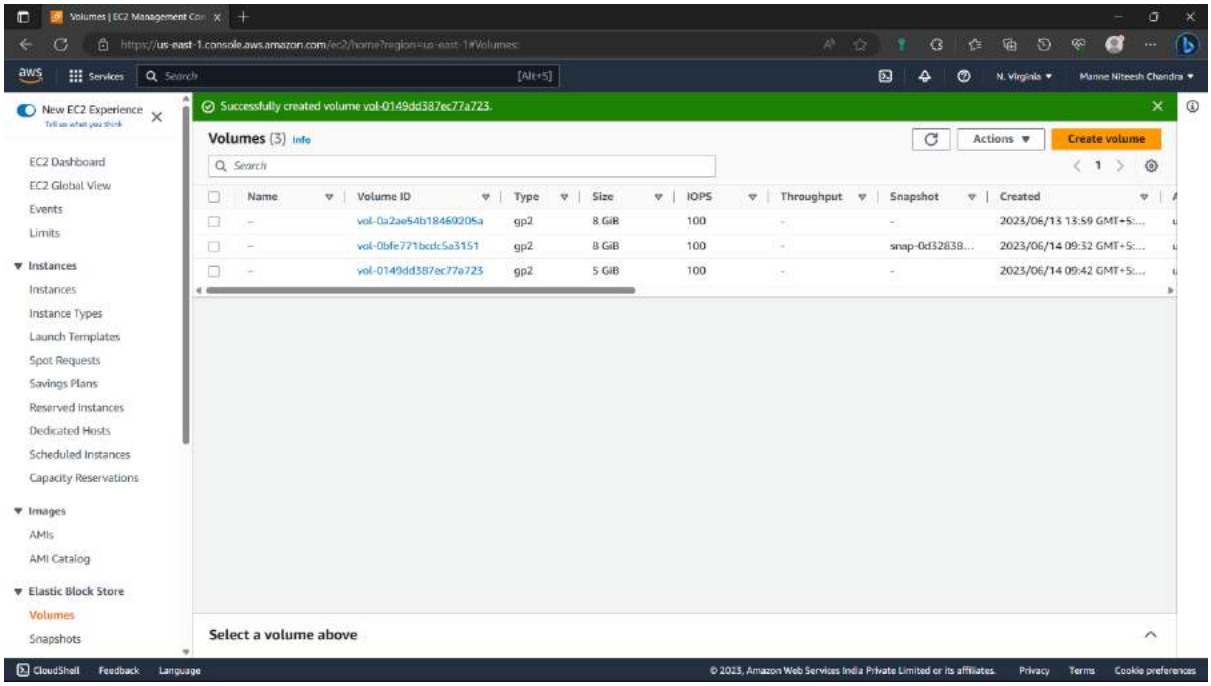
164)Size-5GB, the availability zone of the volume should be in the same zone as instance.



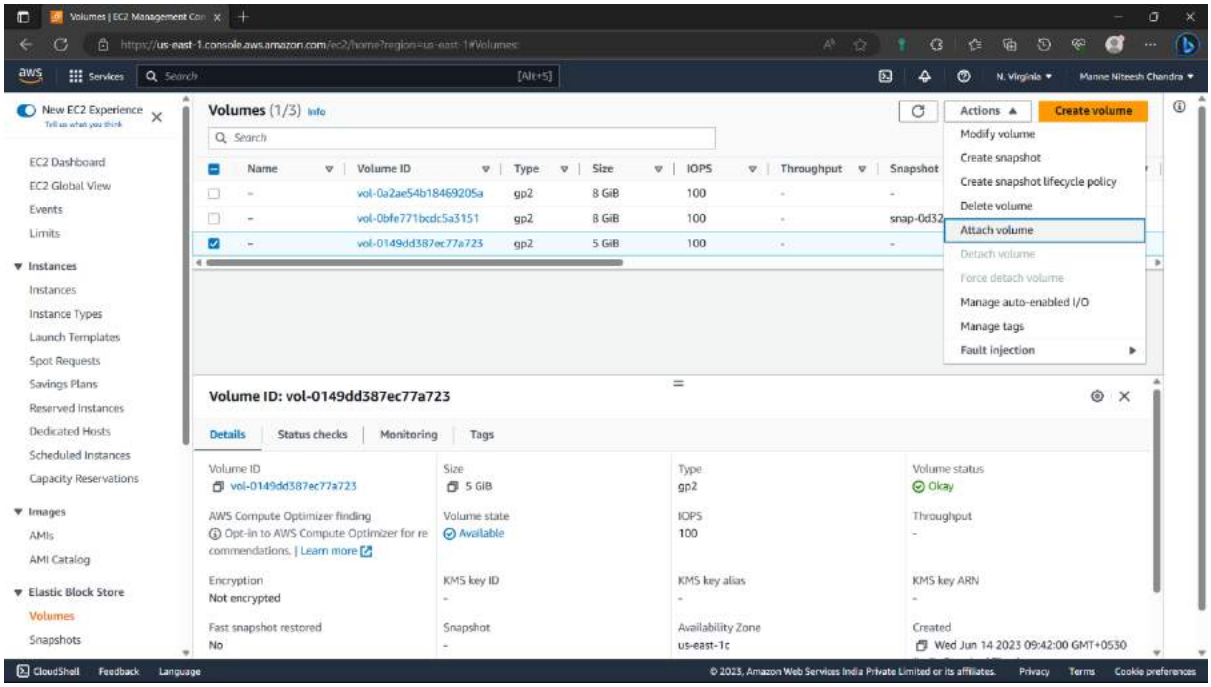
165)you can give tag to this volume i.e., can identify it easily among all the volumes. click on “click volume”.



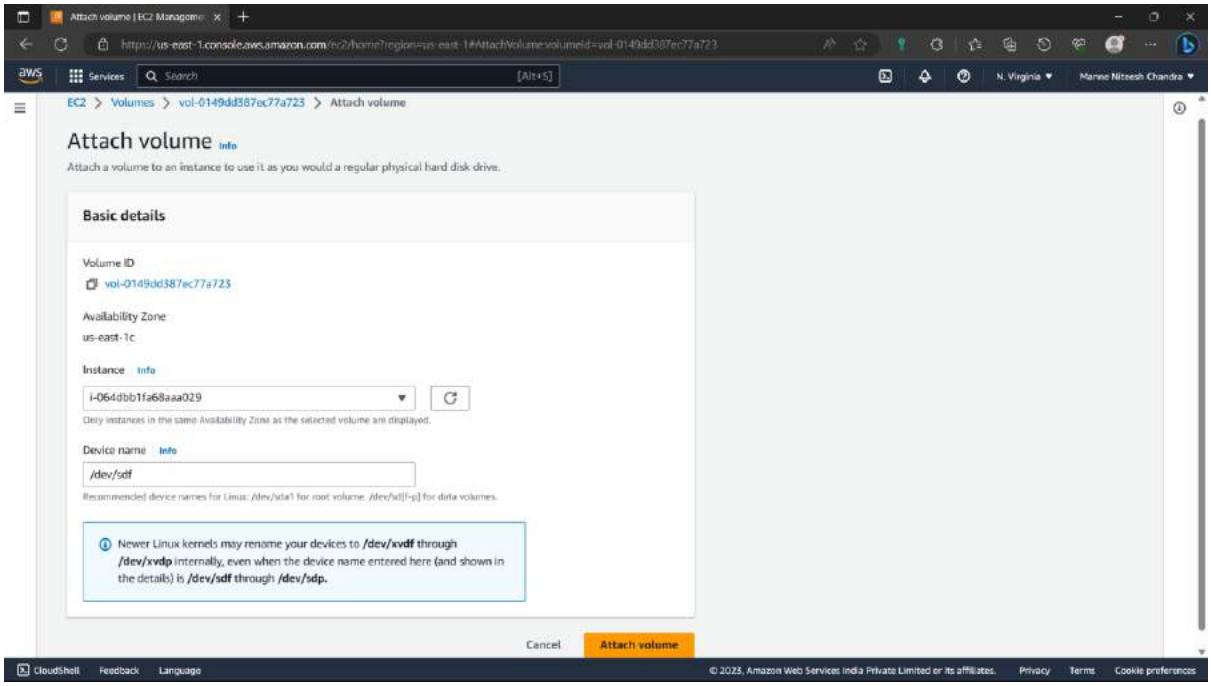
166)If you didn’t give any tag then it would be better to remember the volume Id. The volume is created.



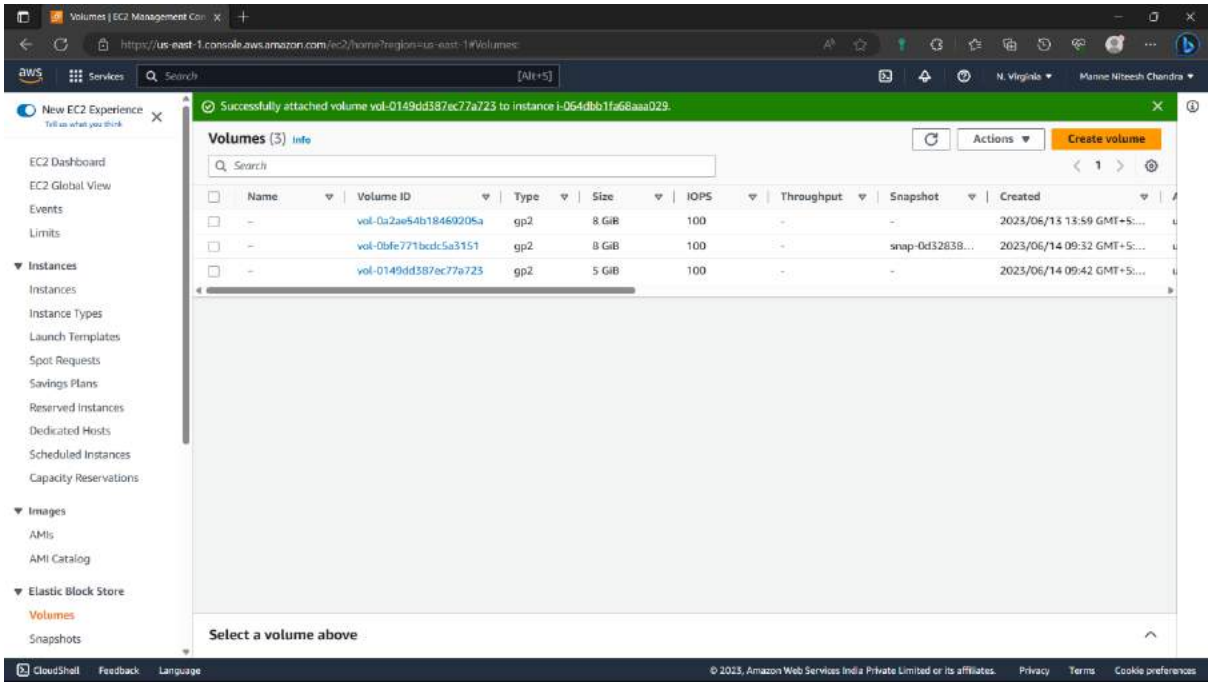
167)Attach this volume to the instance. Refresh the Page and select the volume you created and click on “actions”→”Attach volume”.



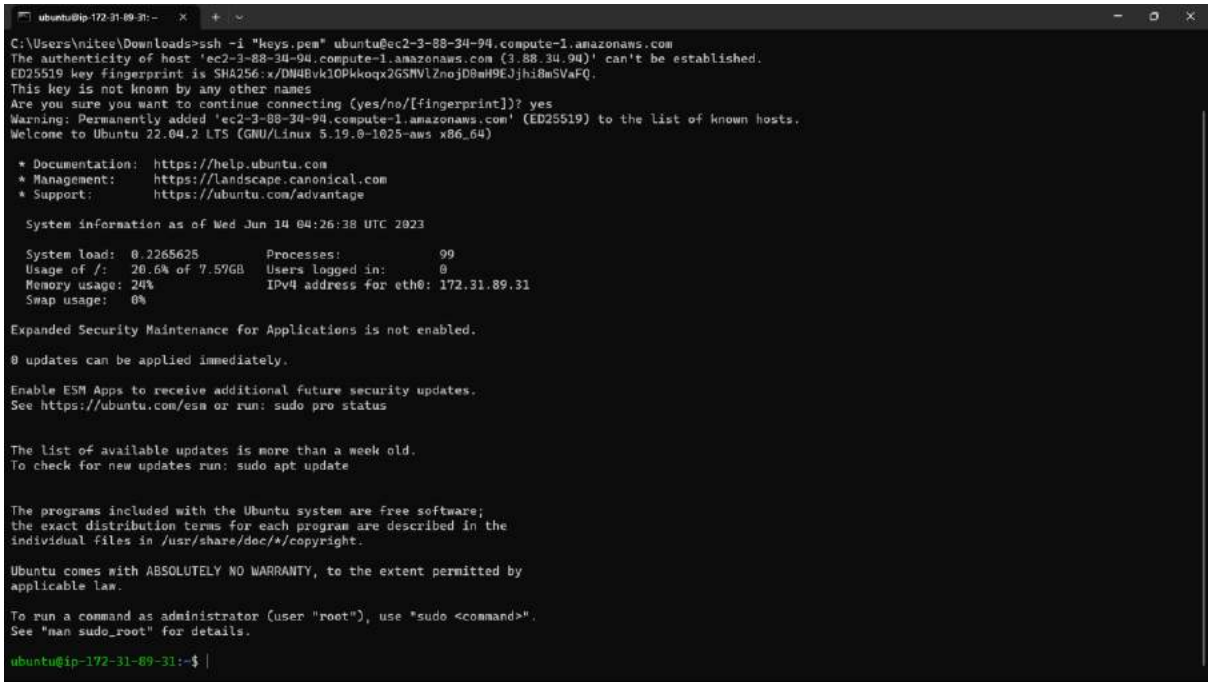
168)Select the instance you want to attach, Click on “Attach volume”.



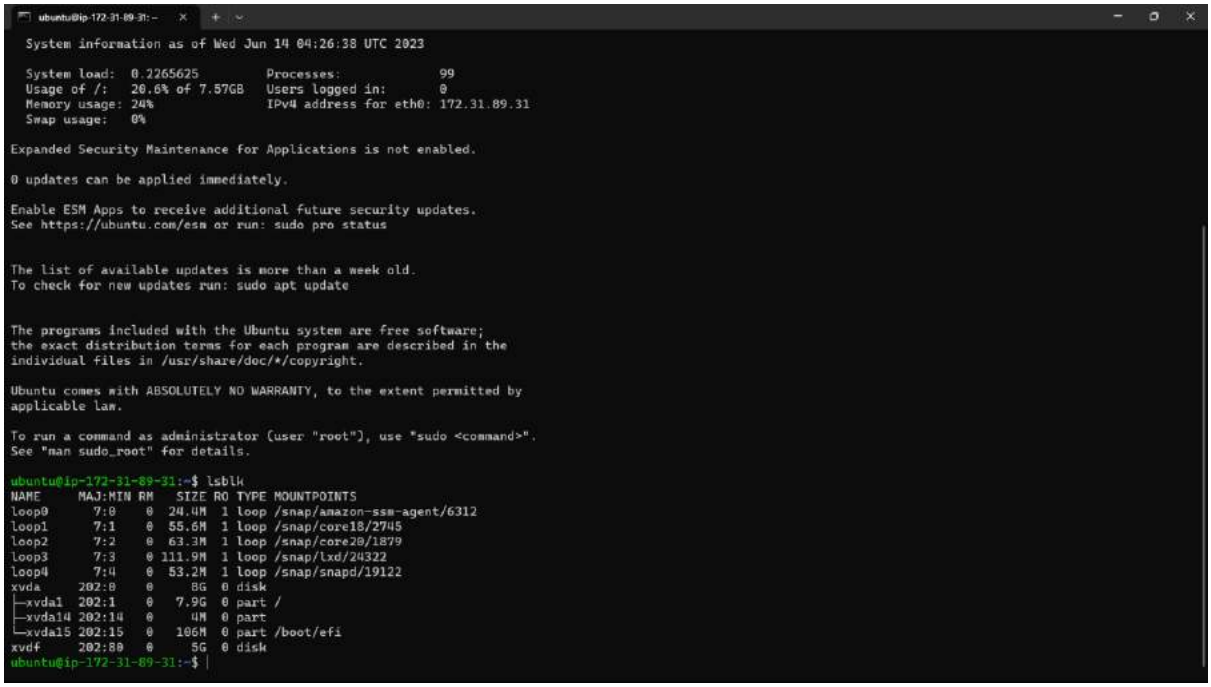
169)The volume is attached to the instance.



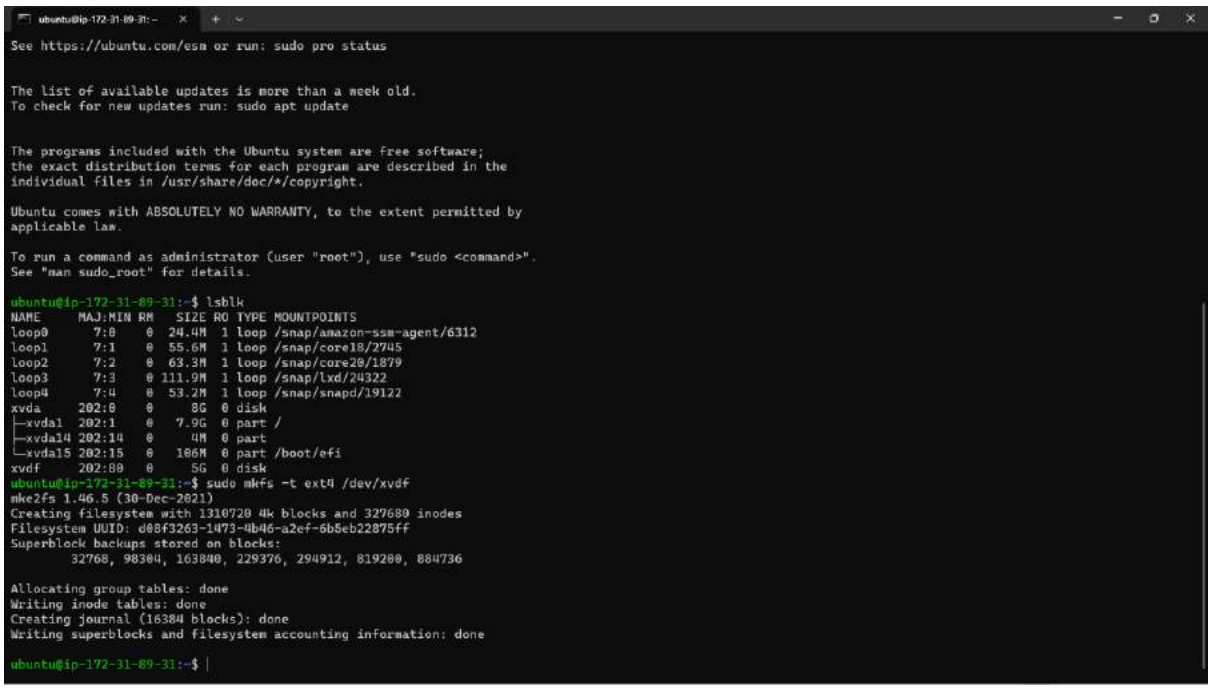
170)Access the instance by following the previous steps: (140-146).



171)Type the command “lsblk”:it will all the volumes attached to this instance. ”xvdf” is the additional volume we attached.



172)Type the command “mkfs -t ext4 /dev/xvdf” if it doesn’t work try using sudo before i.e., “sudo mkfs -t ext4 /dev/xvdf”. It is used to create files in additional volumes.

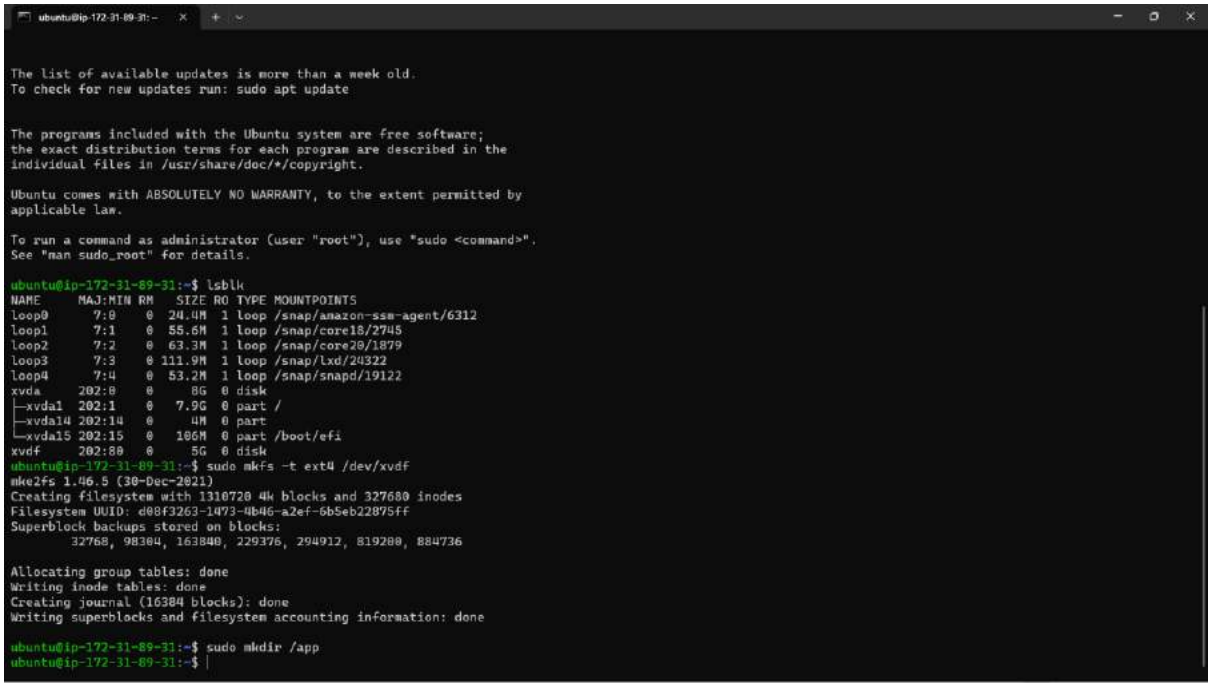


Sudo-super user

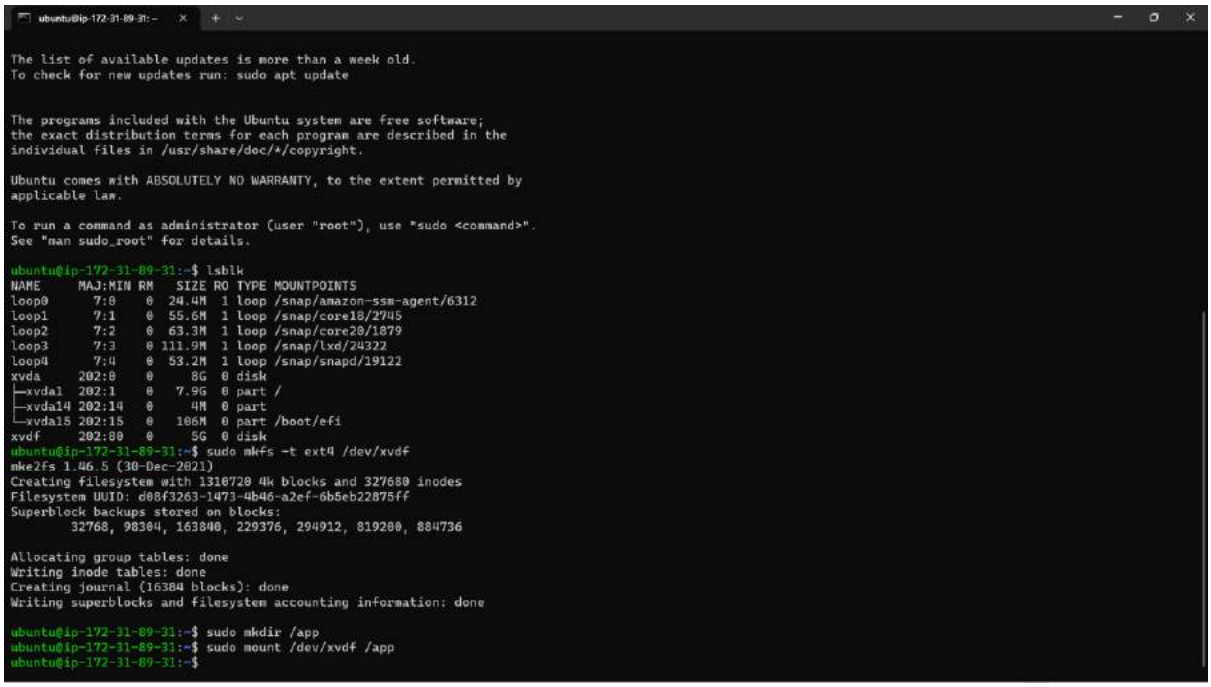
Mkfs-make format.

/dev/xvdf- dev: device, xvdf- additional volume.

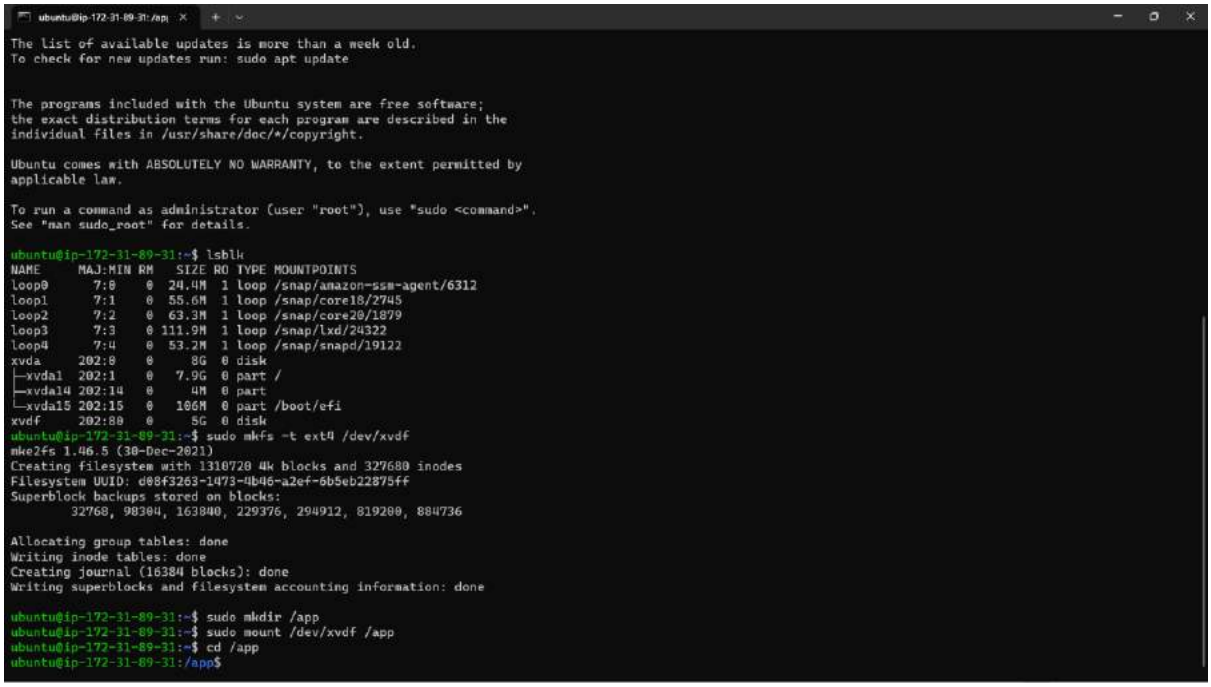
173)Create a repository using the command- “sudo mkdir /reponame” -: “sudo mkdir /app”.



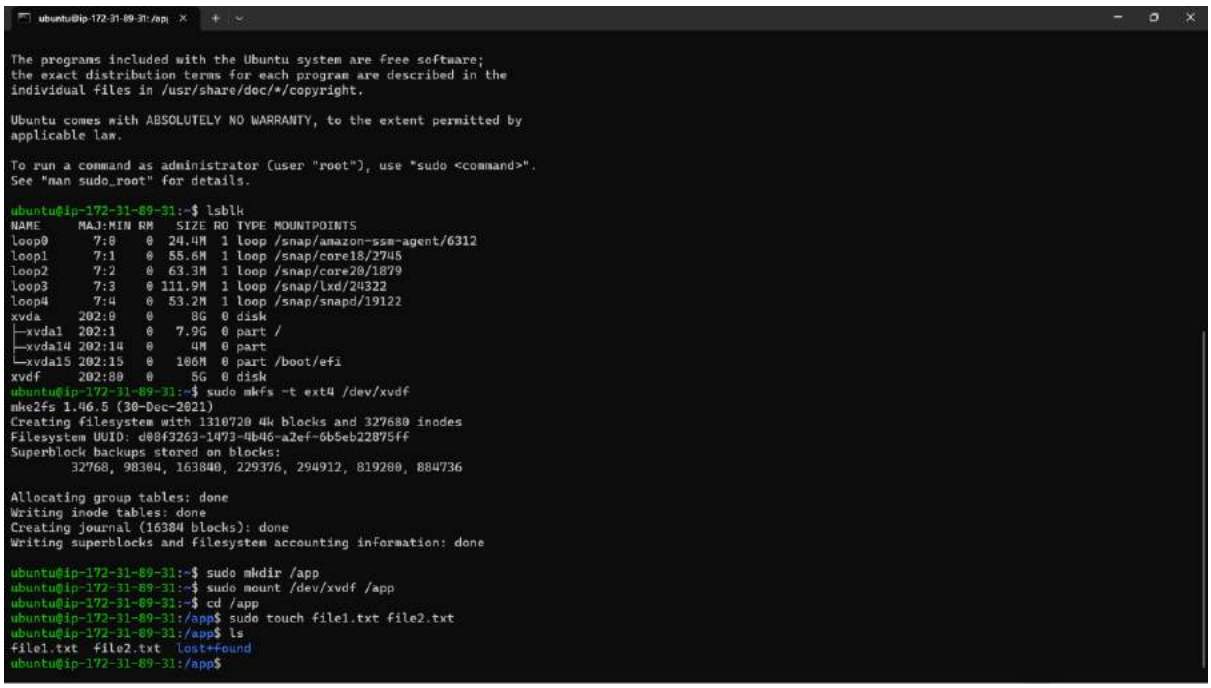
174)Attach the volume to this folder by using the command : “sudo mount /dev/xvdf /app”.



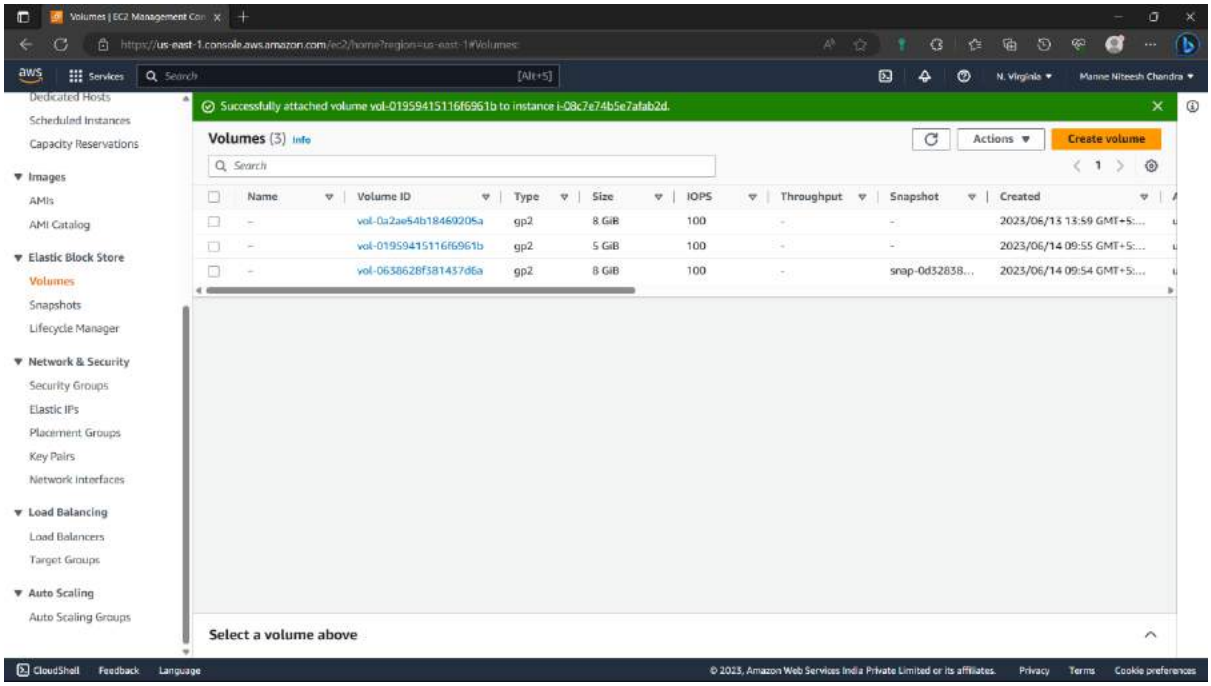
175)Navigate to that folder by using the command – “cd foldername” -: “cd /app”.



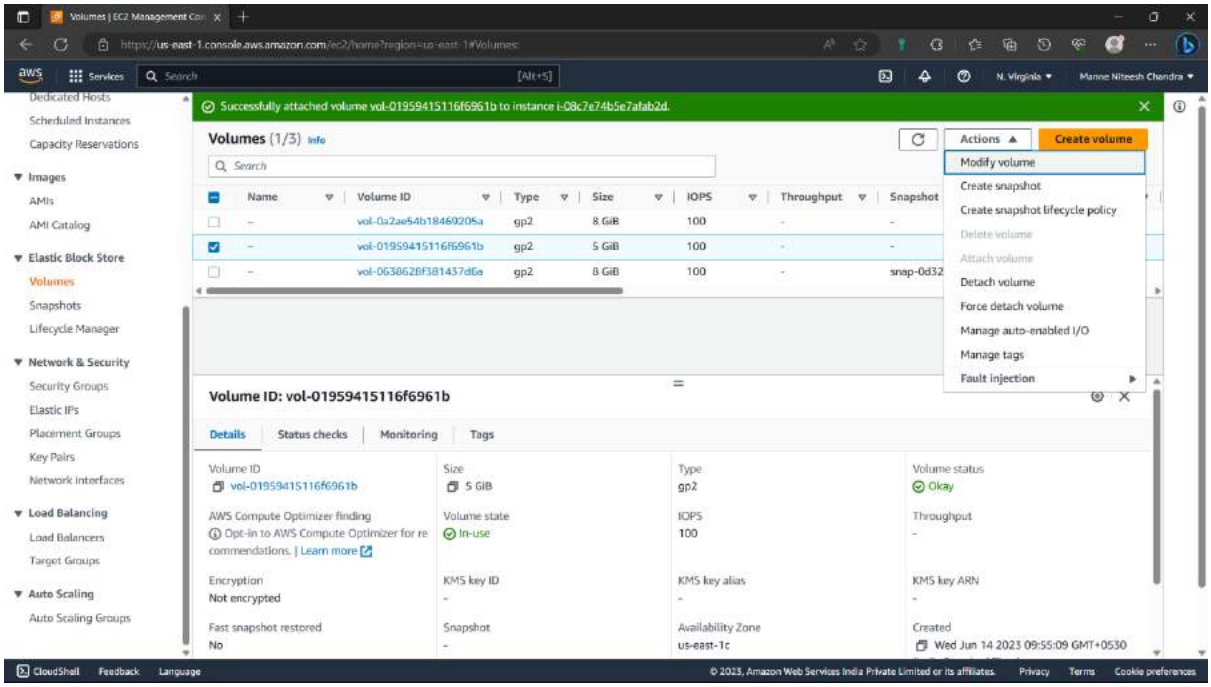
176)Create two files inside the colder using the command- “sudo touch filename” -: “sudo touch file1.txt file2.txt”. you can check the files created or not using the command “ls”.



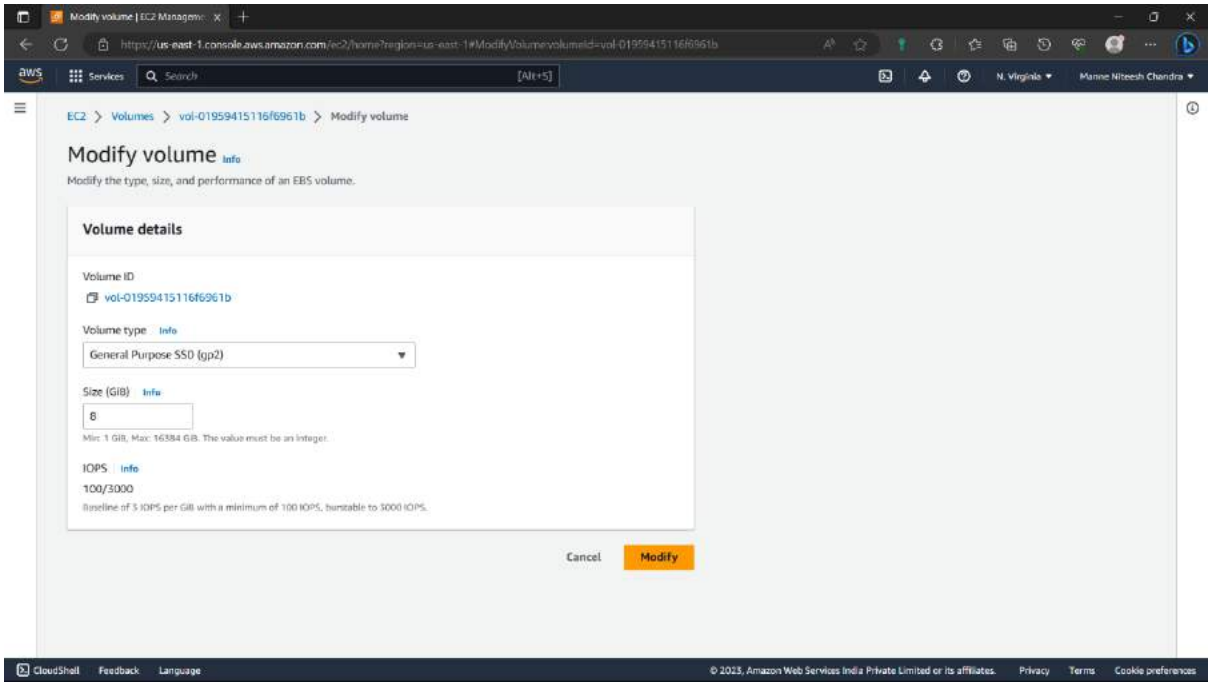
177)Now go to the volumes in the ec2.



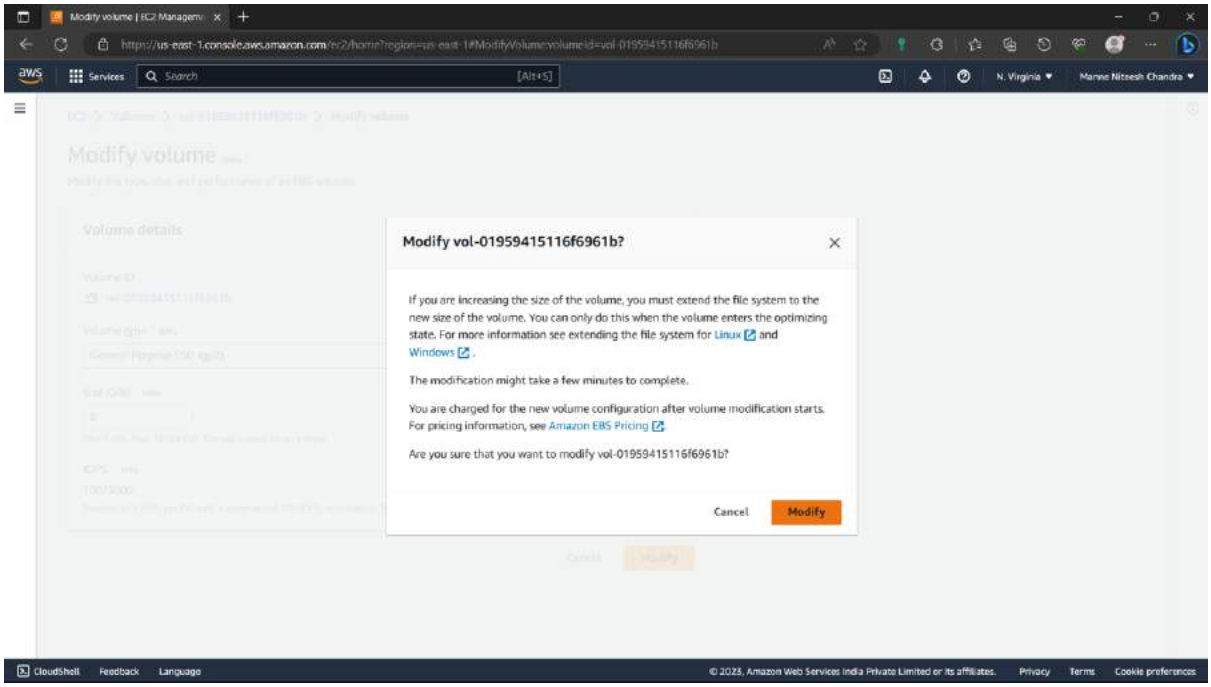
178)Select the additional volume and click on “actions”→”modify volume”.



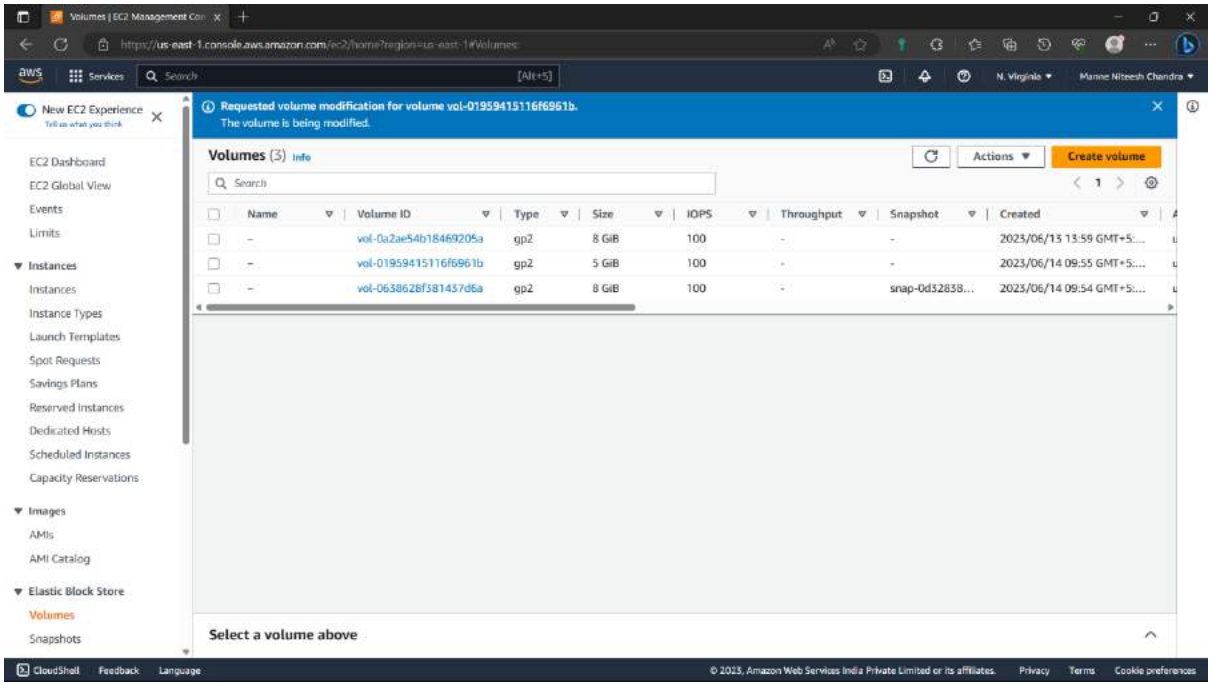
179)Increase the size to 8GB and click on “Modify”.



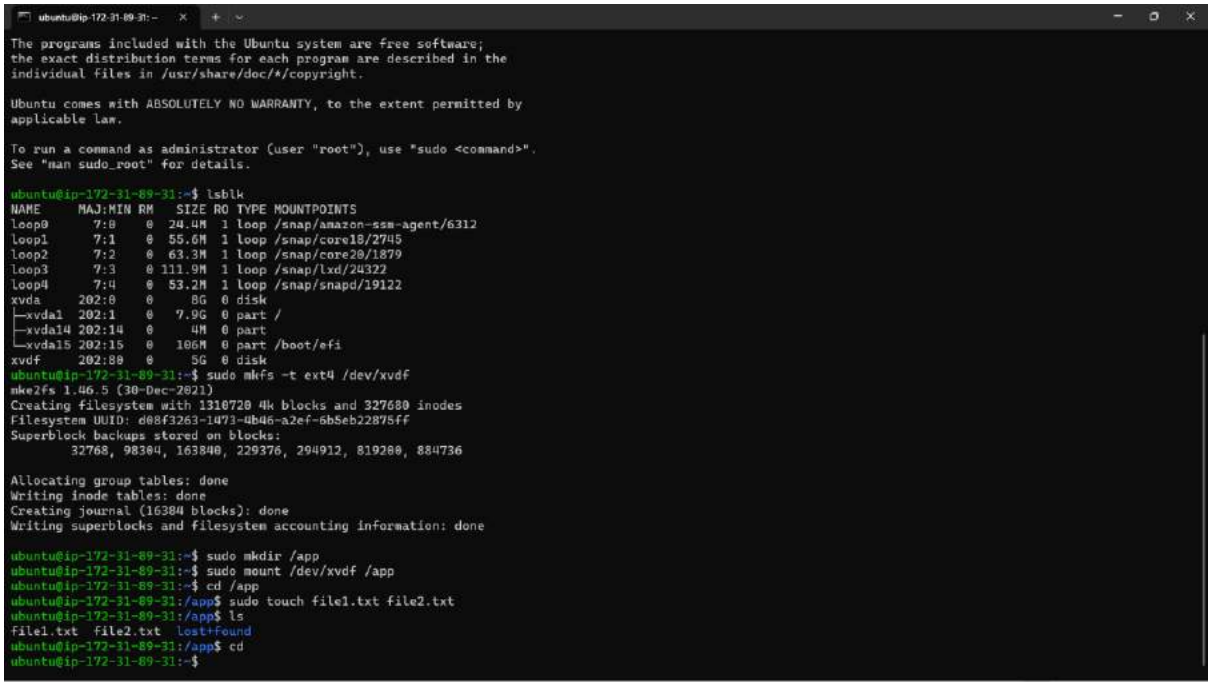
180)Click on modify again.



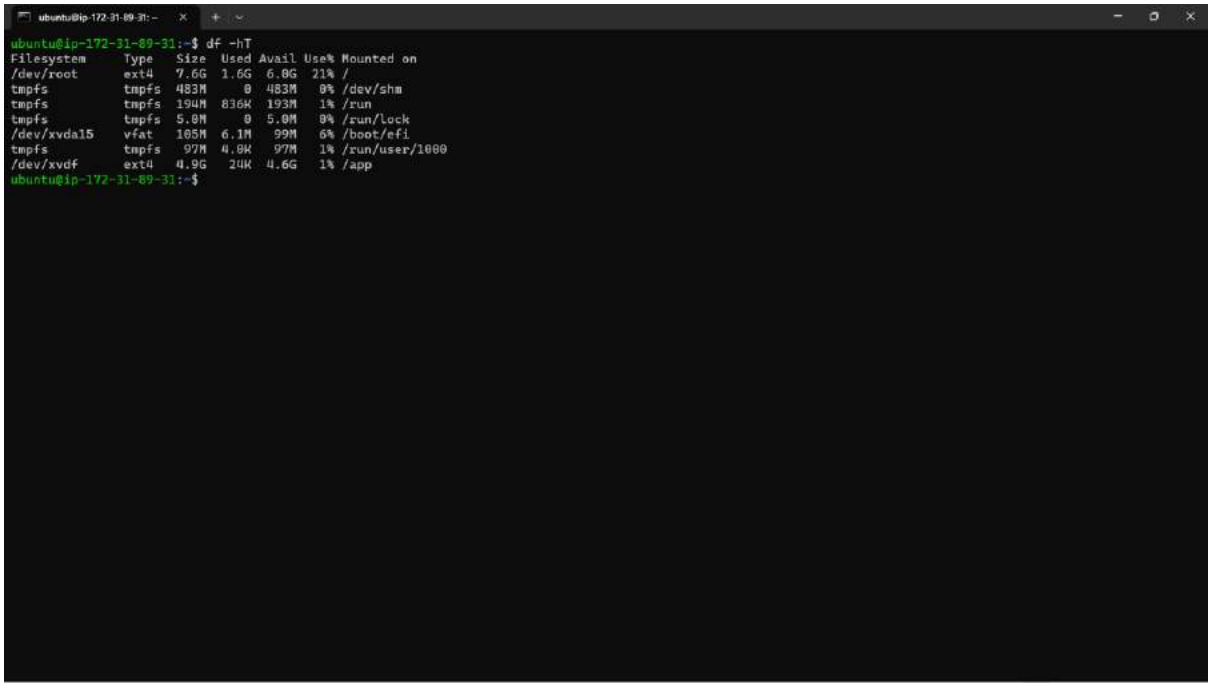
181)Now the request in made.



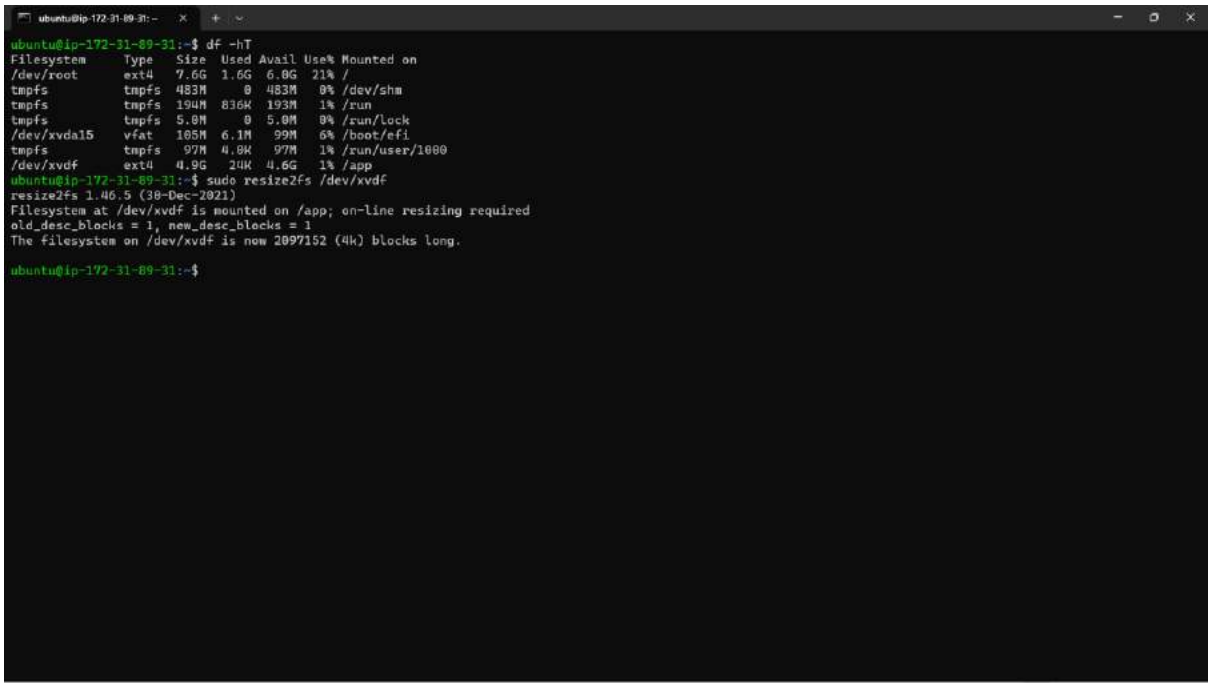
182)Go to your instance type the command “cd”:-will bring back to the previous path.



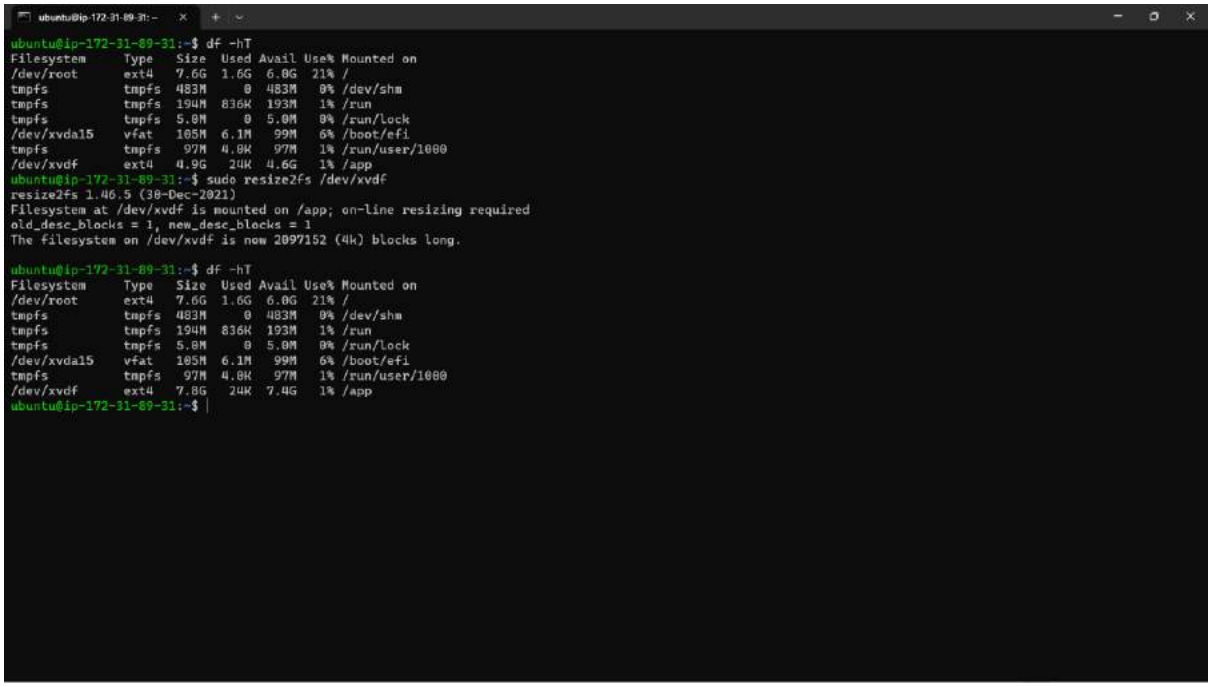
183)Type the command “df -hT”: shows name, size, type, and mount point for the file system that you need to extend.



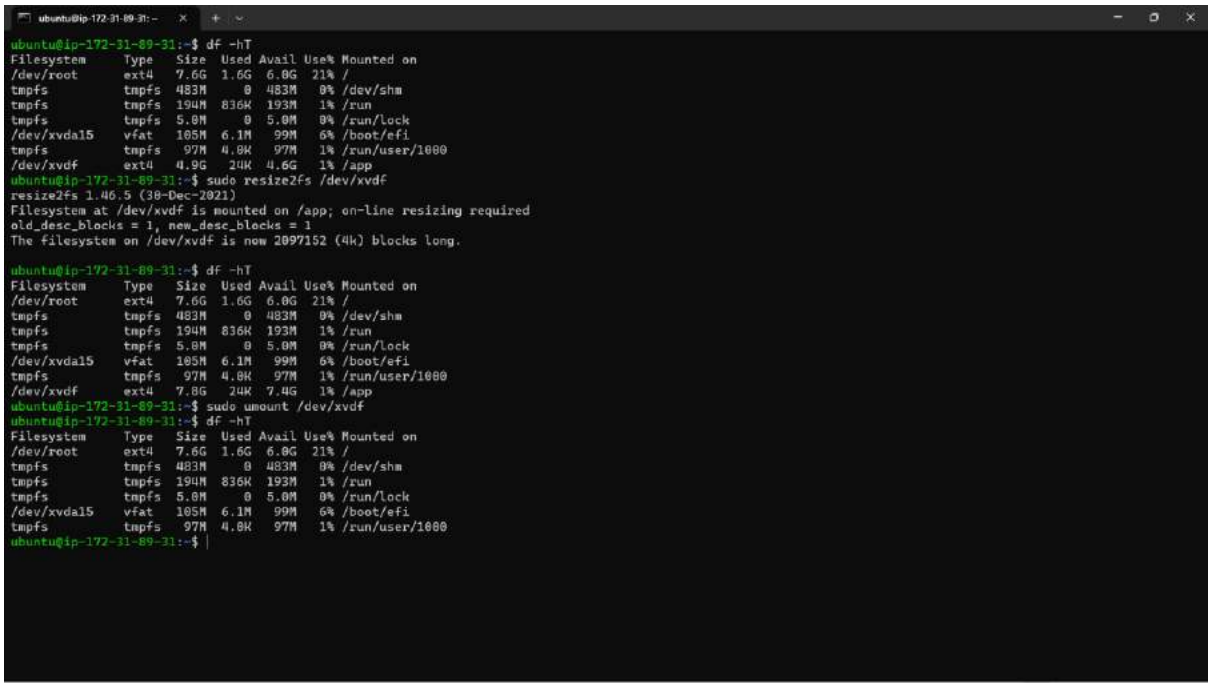
184)File system type is “Ext4” then the command is “sudo resize2fs /dev/filesystem” -: “sudo resize2fs /dev/xvdf”.



185)Type the command: “df -hT” to check the updated storage, we can see that the additional volume xvdf changes form 5gb→8gb.

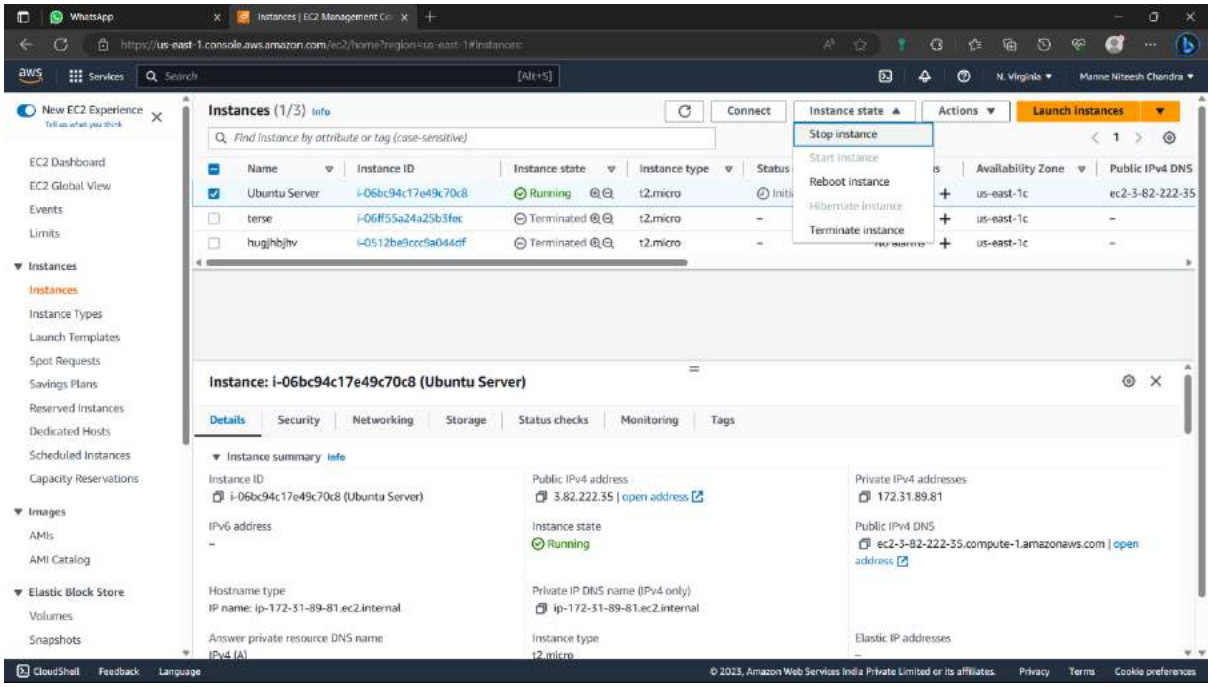


186)Now detach the volume from this instance by using the command : “sudo umount /dev/devicename” -; “sudo umount /dev/xvdf”. We can check it by using the command “df -hT”.

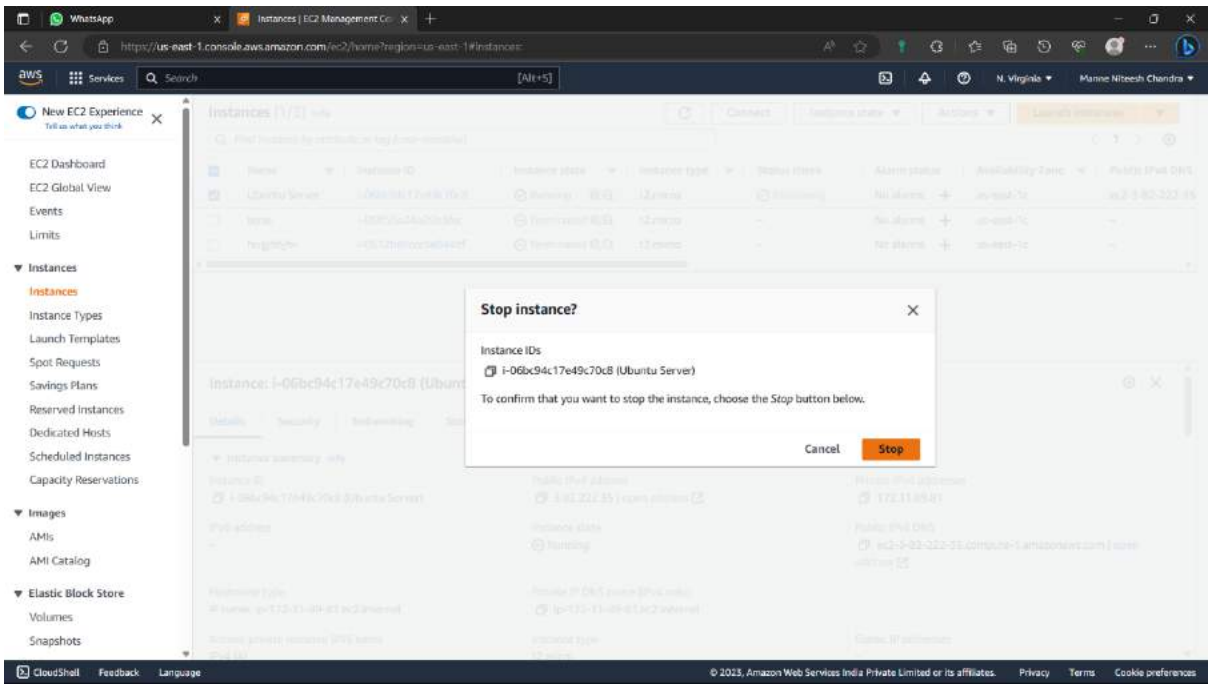


We cannot see the “/dev/xvdf”because it is detached from the instance.

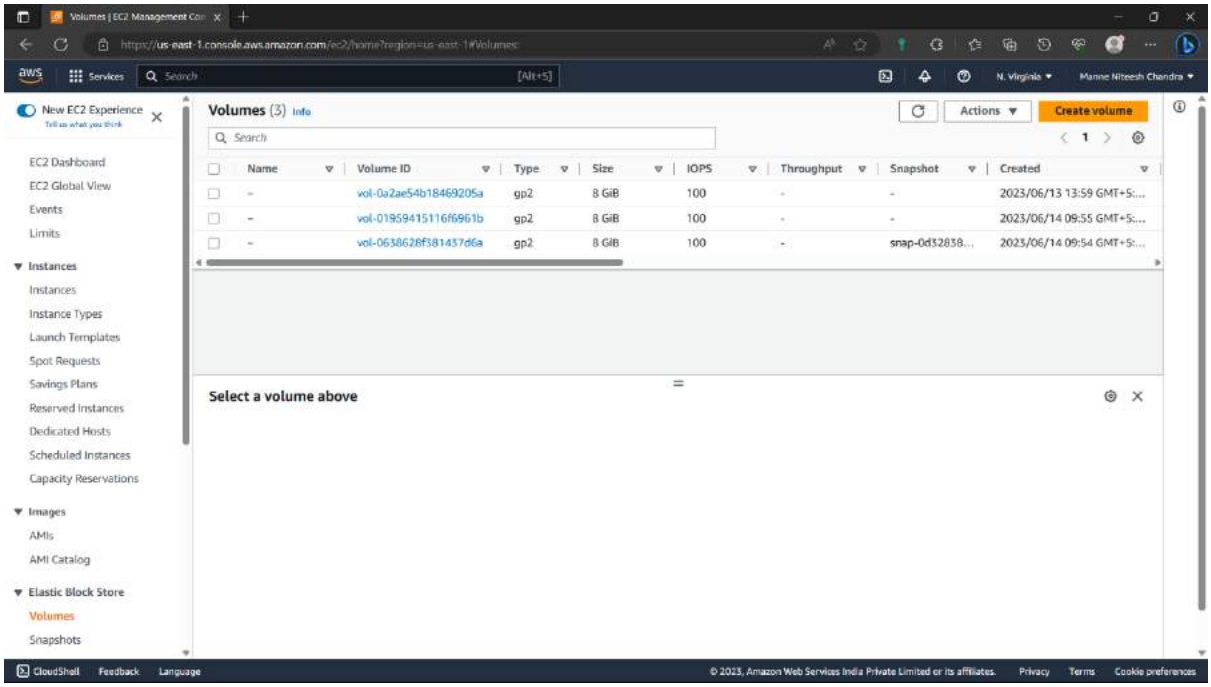
187)Select the instance i.e., you created(Additional volume attached)and click “instance state”→”Stop instance”.



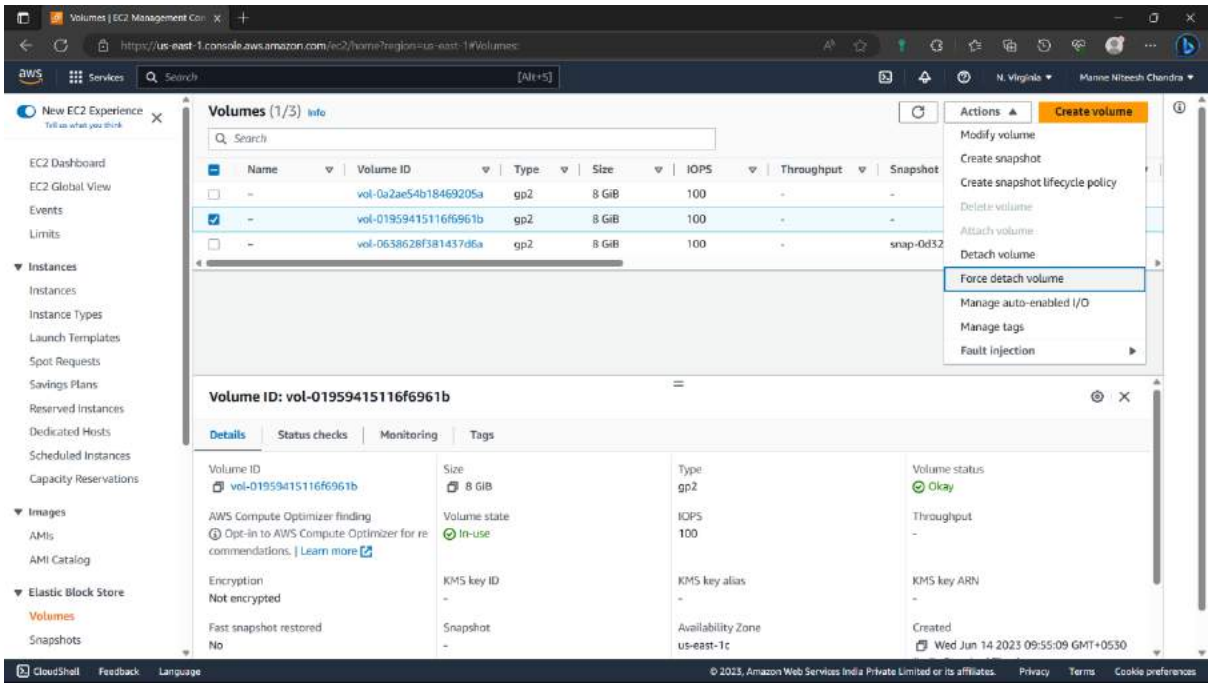
188)Click on “stop”.



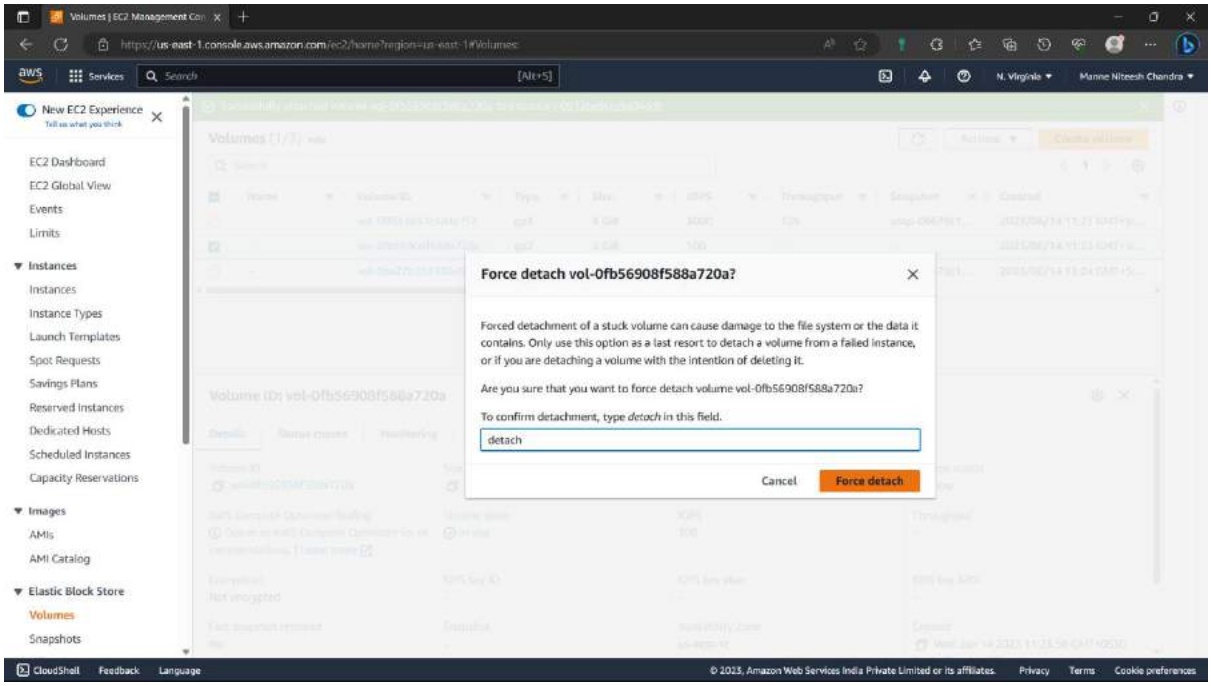
189)Navigate to “volumes” in the Elastic block storage at the bottom of the navigation panel.



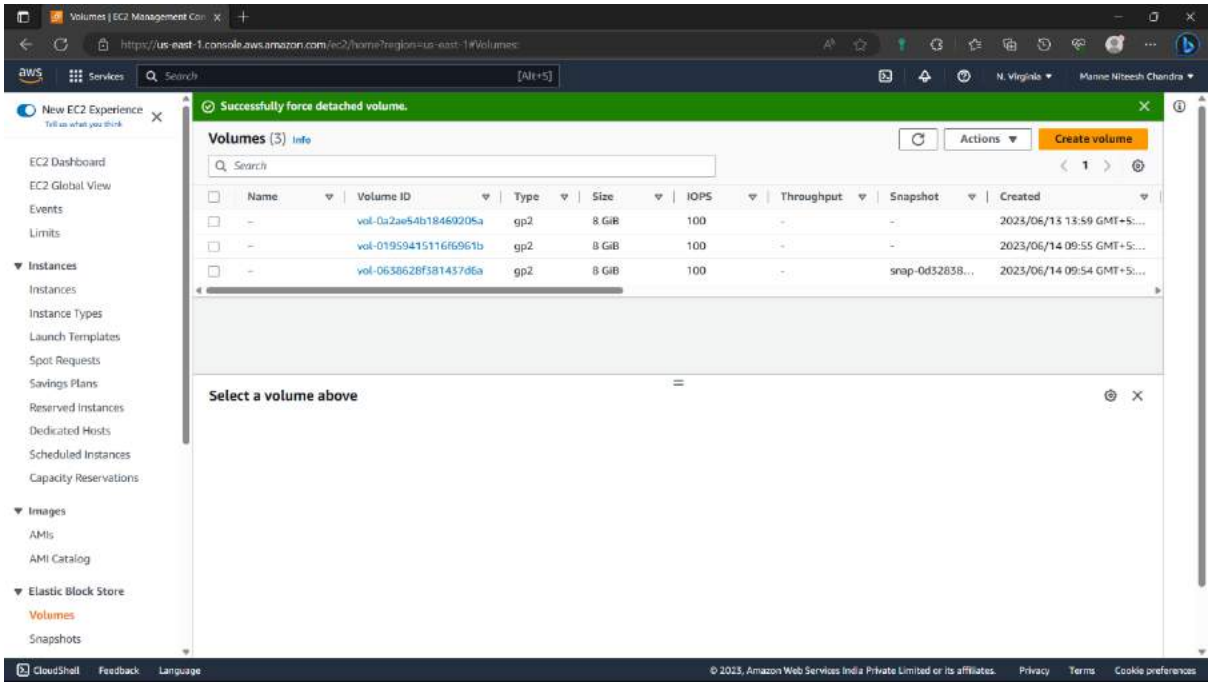
190)Select the additional volume by volume id or by the tag(if you given). Click on “Actions”→”Force detach volume”. Because normal detach take so much time so we use force detach.



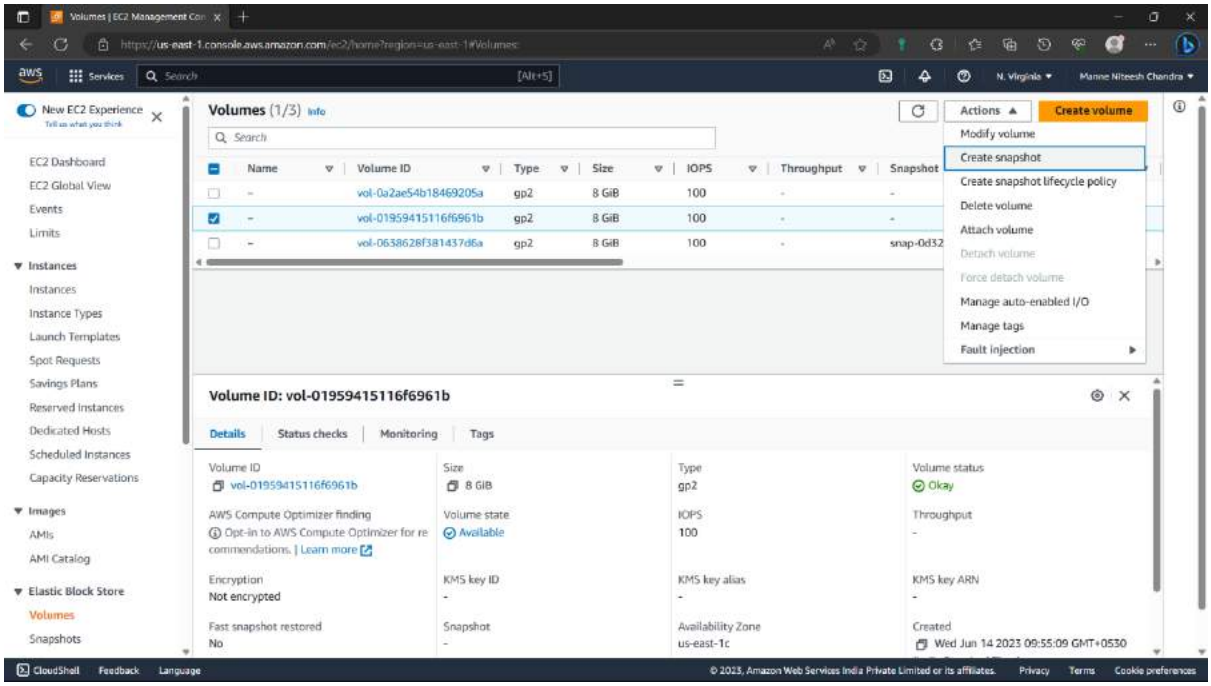
191)Type “detach” and click on “force detach” option.



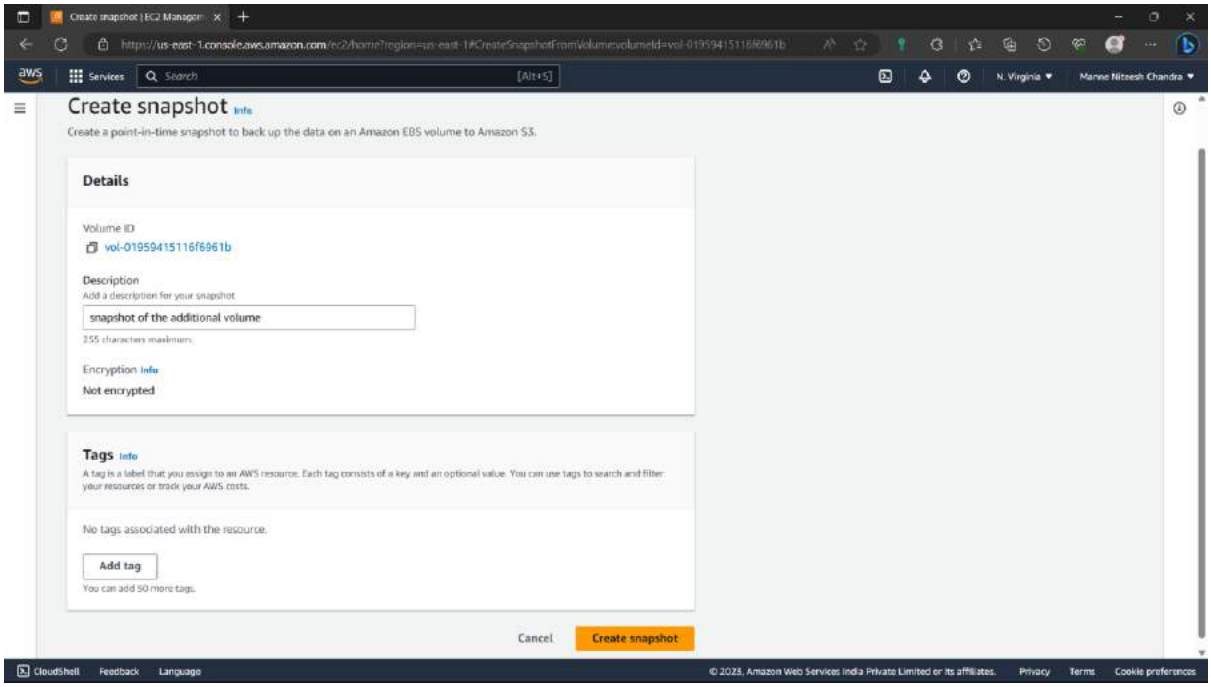
192)Now the volume is detached successfully.



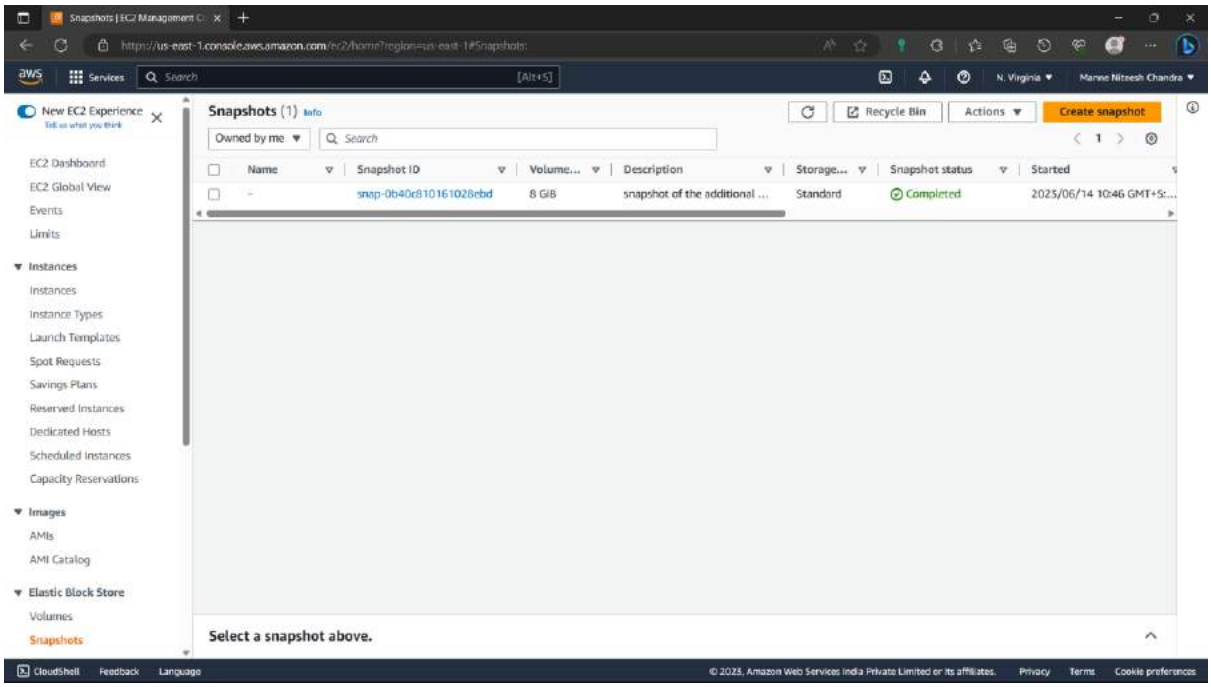
193)Select the additional volume click on “actions”→”create snapshot”



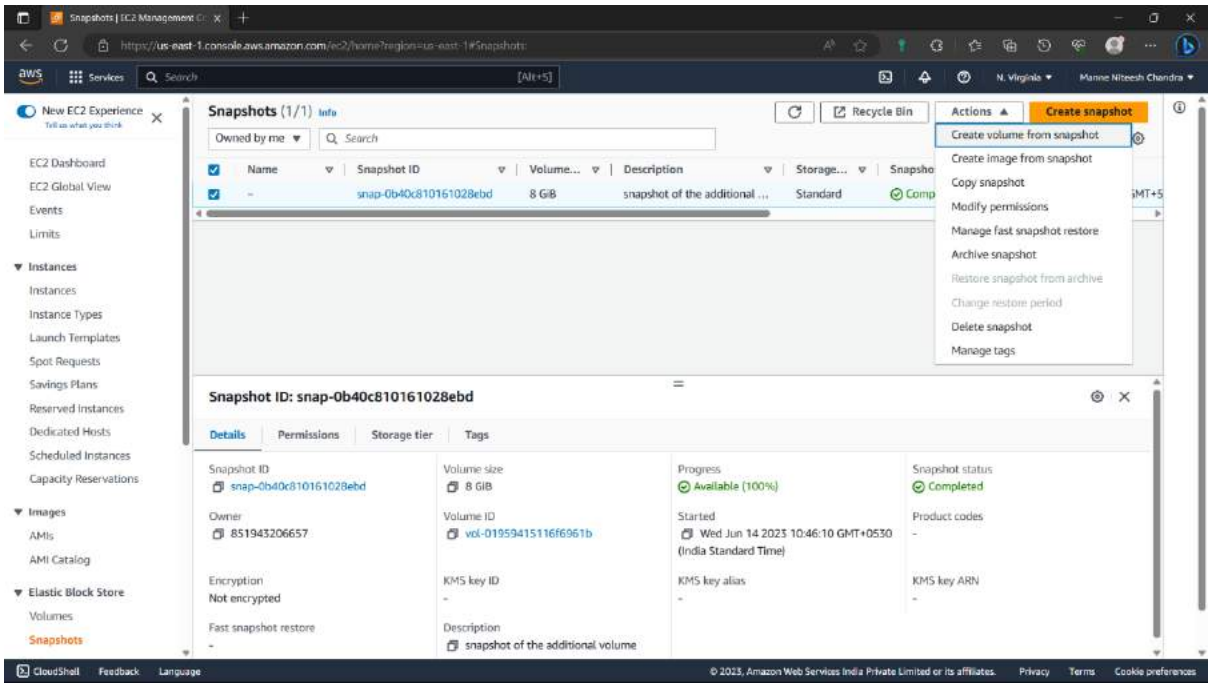
194)Give a name if you want by clicking on “add tag”. Click on “Create snapshot”.



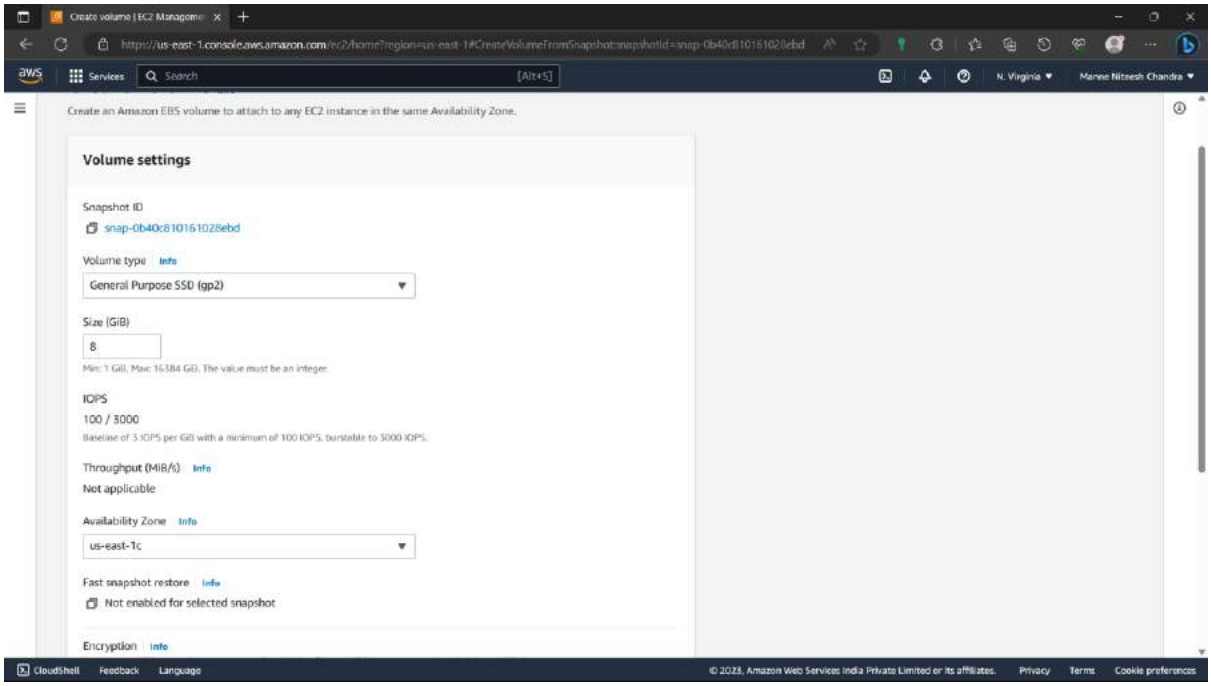
195)Now the snapshot is created you can check it by going to “snapshot” below the “volumes” in “Elastic block storage” on the navigation panel.



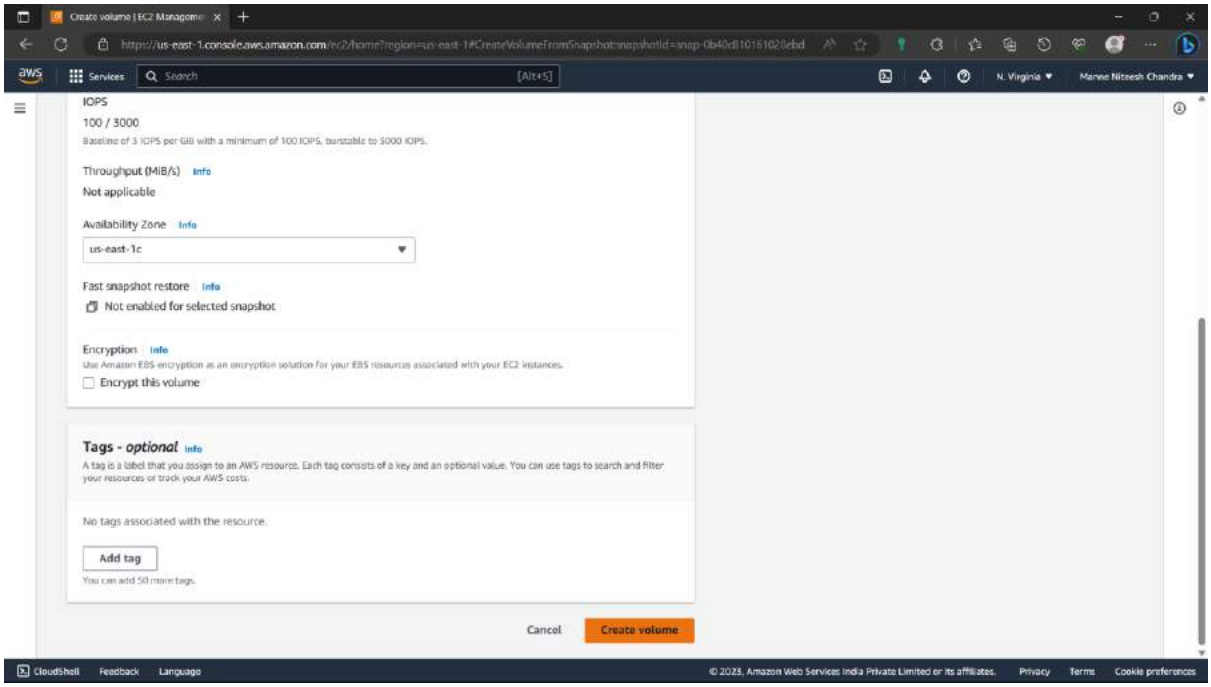
196)Select the snapshot click on “actions”→”create volume from snapshot”.



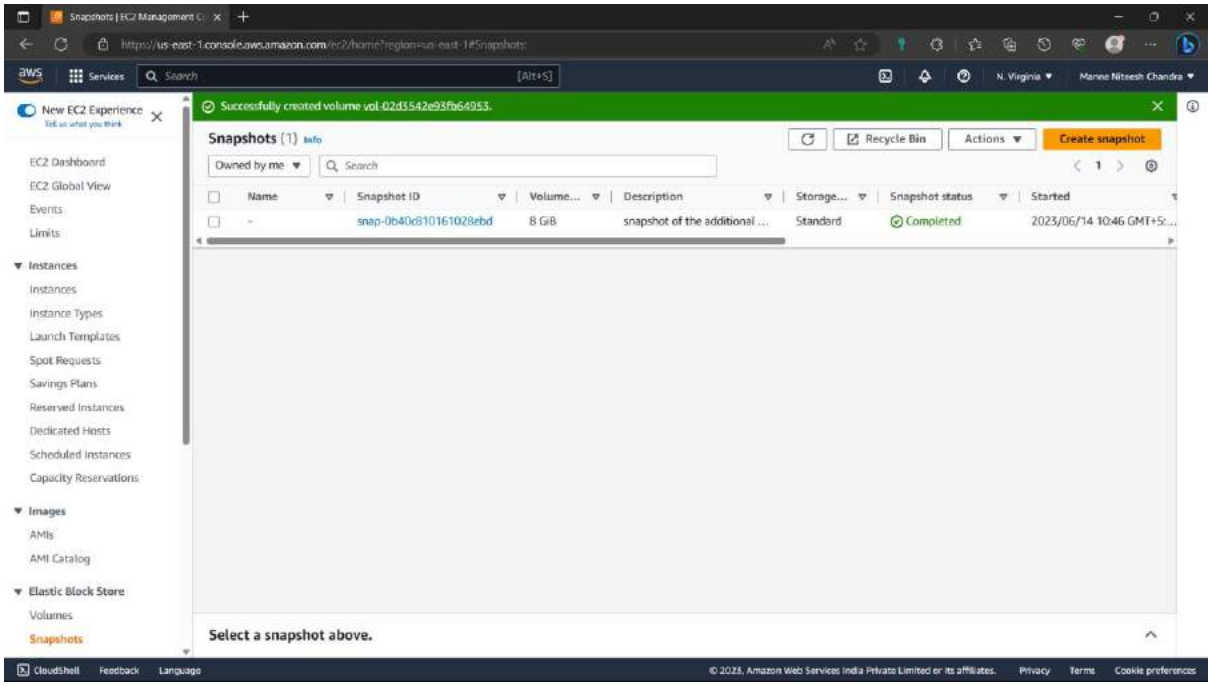
197)The availability zone should be same as instance availability zone.



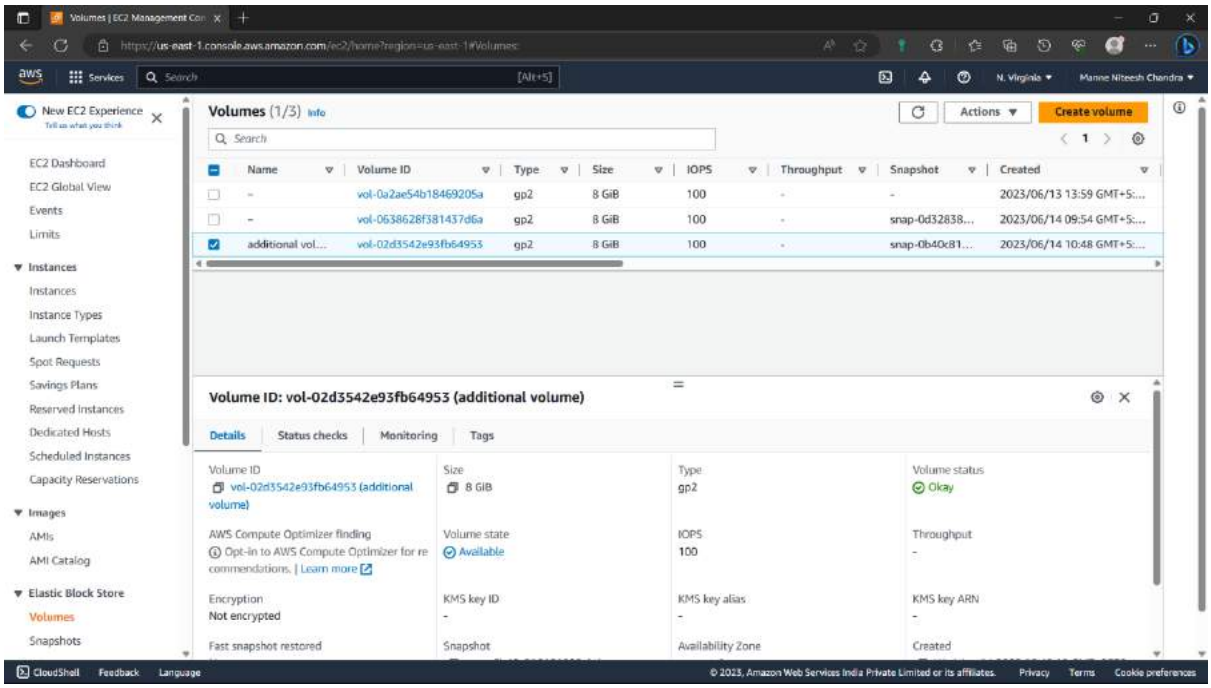
198)Click on “create volume”.



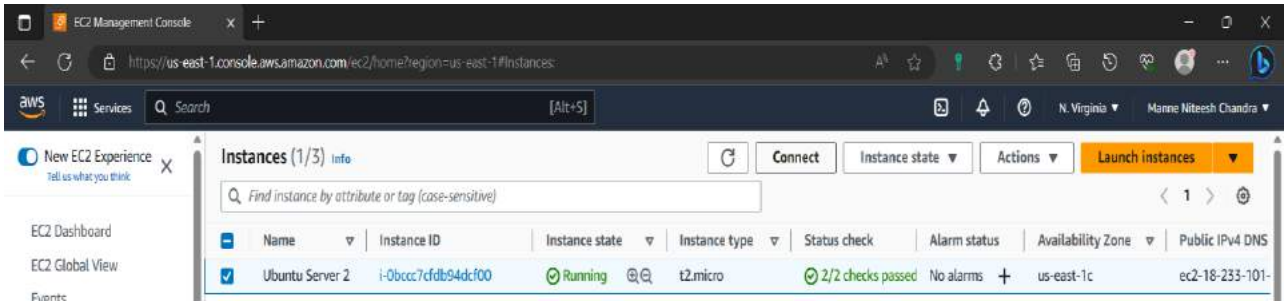
199)Now the volume is created from the snapshot.



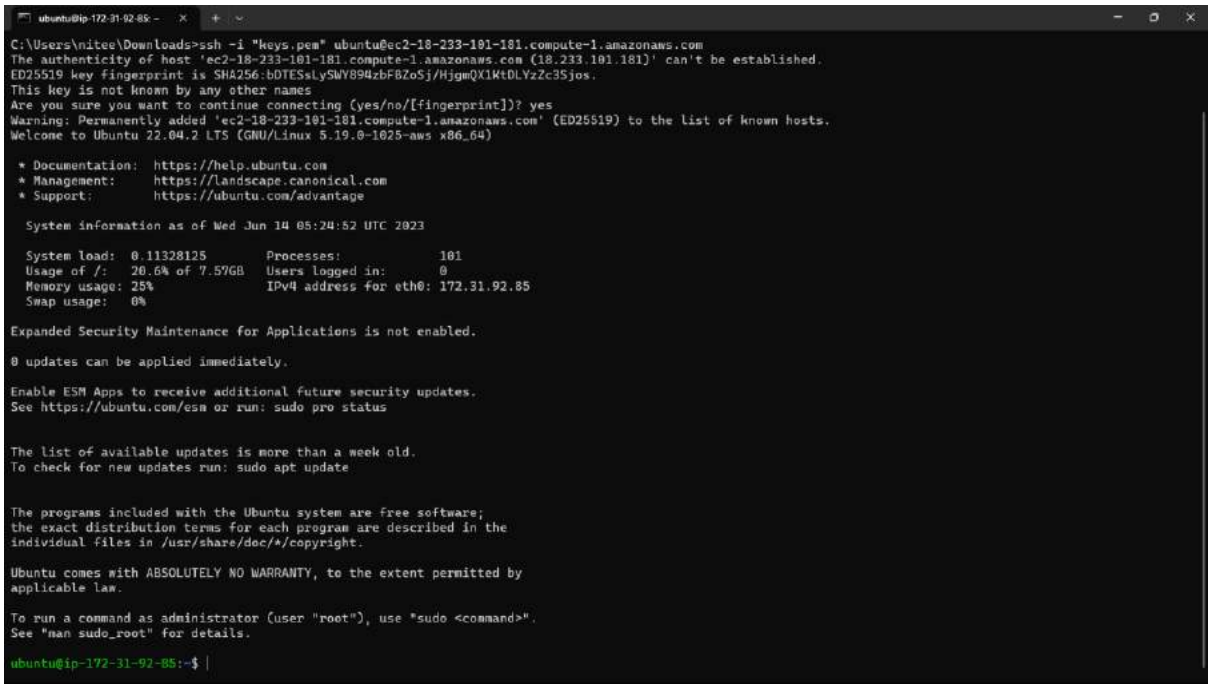
200)Give a name to the additional volume so that you can identify it easily by clicking on the additional volume at “name” column. after typing the name click on save now the name is saved for the additional volume, identify it easily



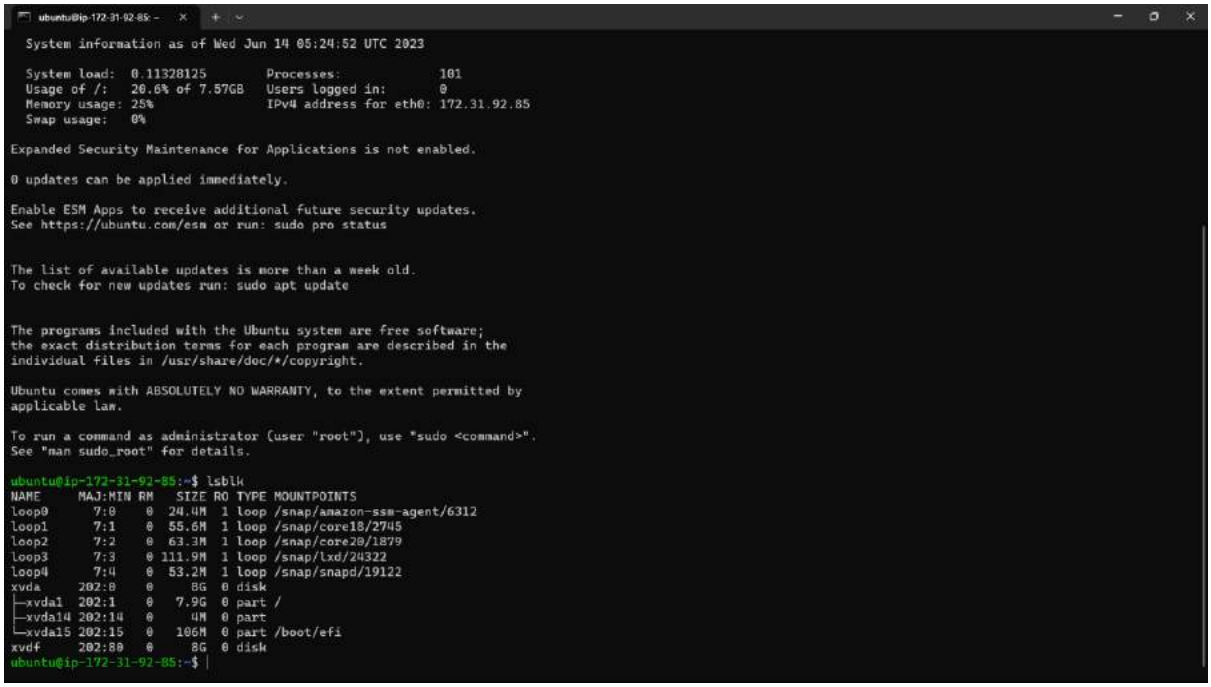
201)Create a new instance by following the steps:129-139.



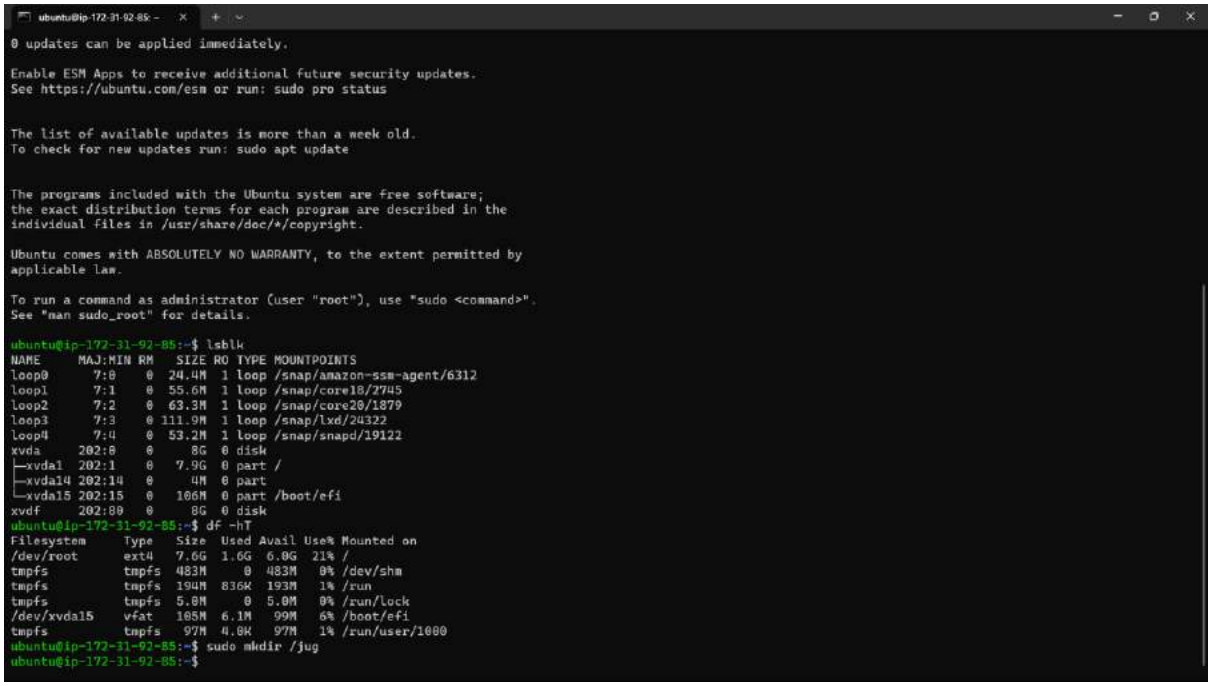
202)Access the instance by following the steps:(140-146).



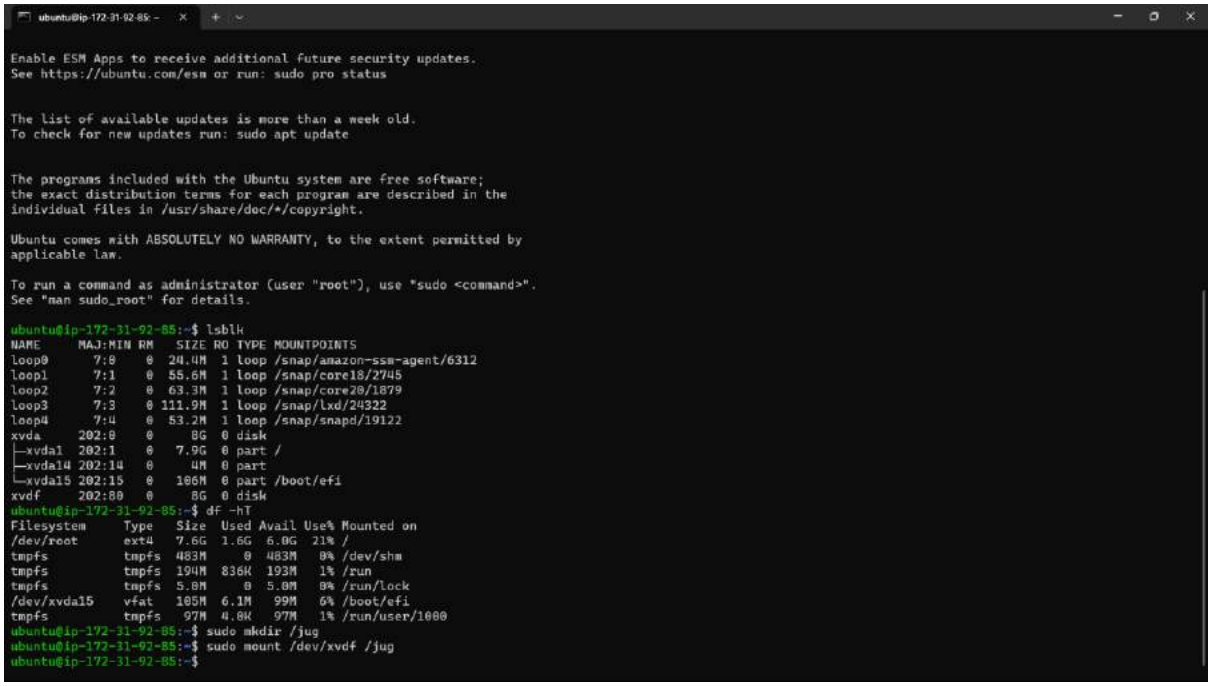
203)Type the command “lsblk” to see the attached voulmes we can see the additional attached volume .



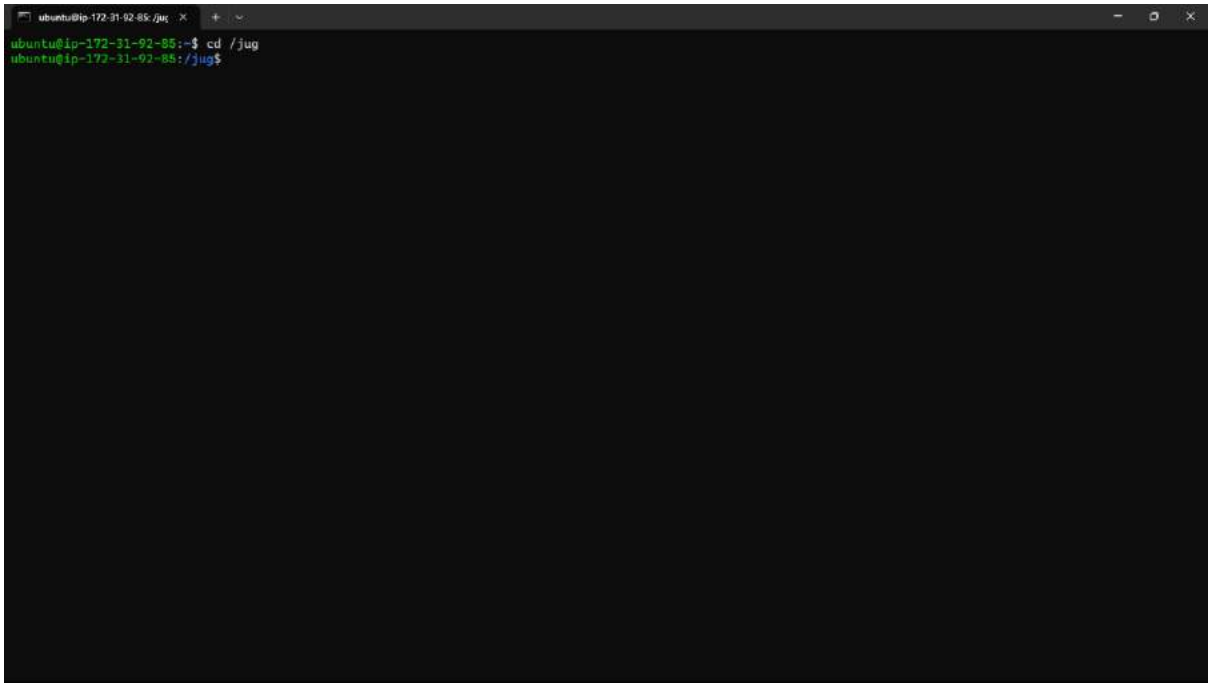
204)Create a repository using the command “sudo mkdir reponame” – “sudo mkdir /jug”.



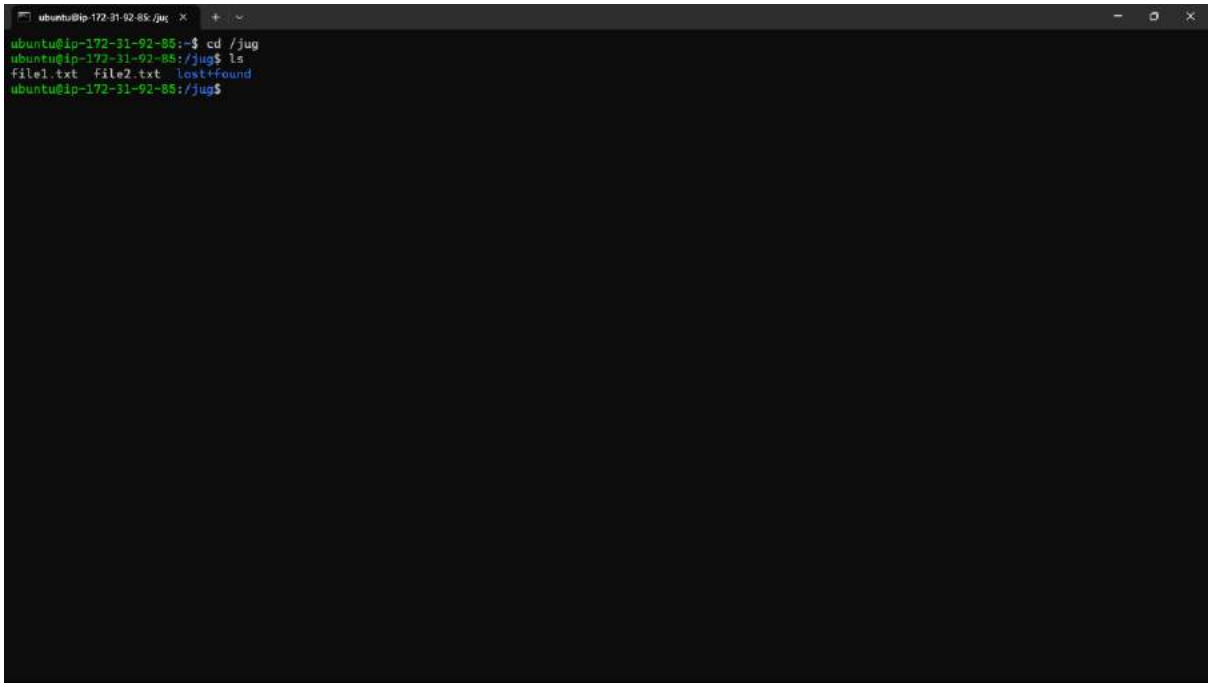
205)Now attach the volume to these repository – “sudo mount /dev/xvdf /jug”.



206)Go to the repository using the command “ cd reponame” -: “cd /jug”.



207)Type the “ls” command, we can see the list of files we created in the first mount.

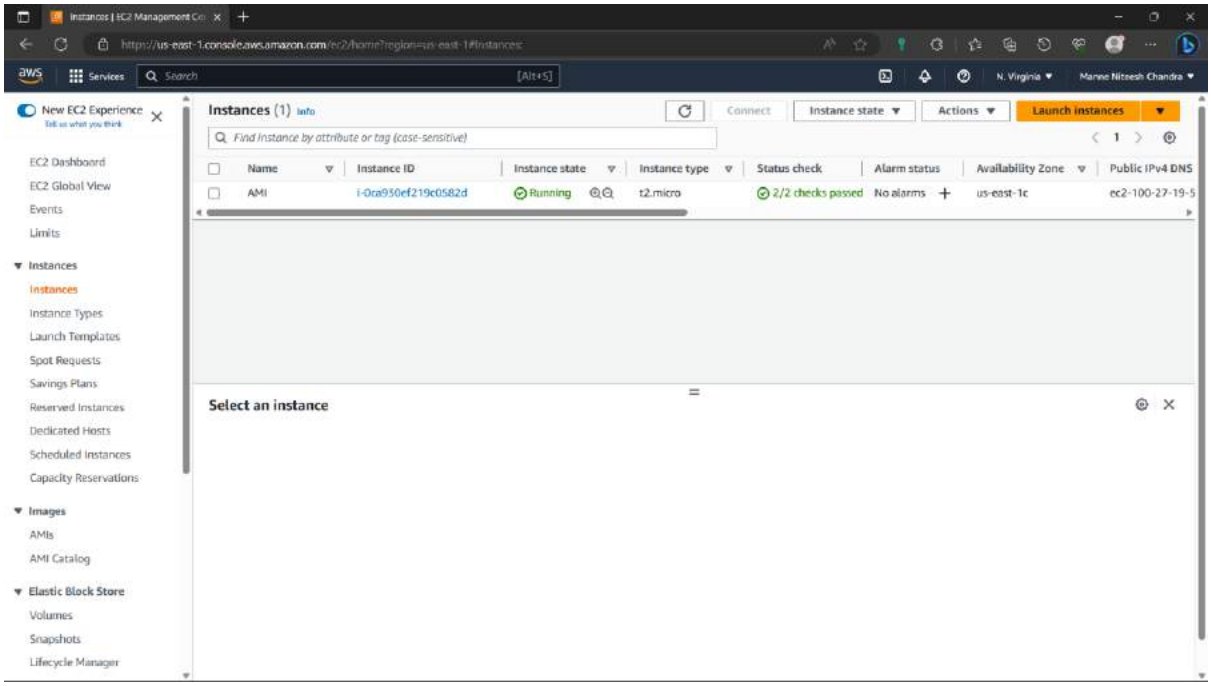


EBS-It is used to share the data in same availability zones

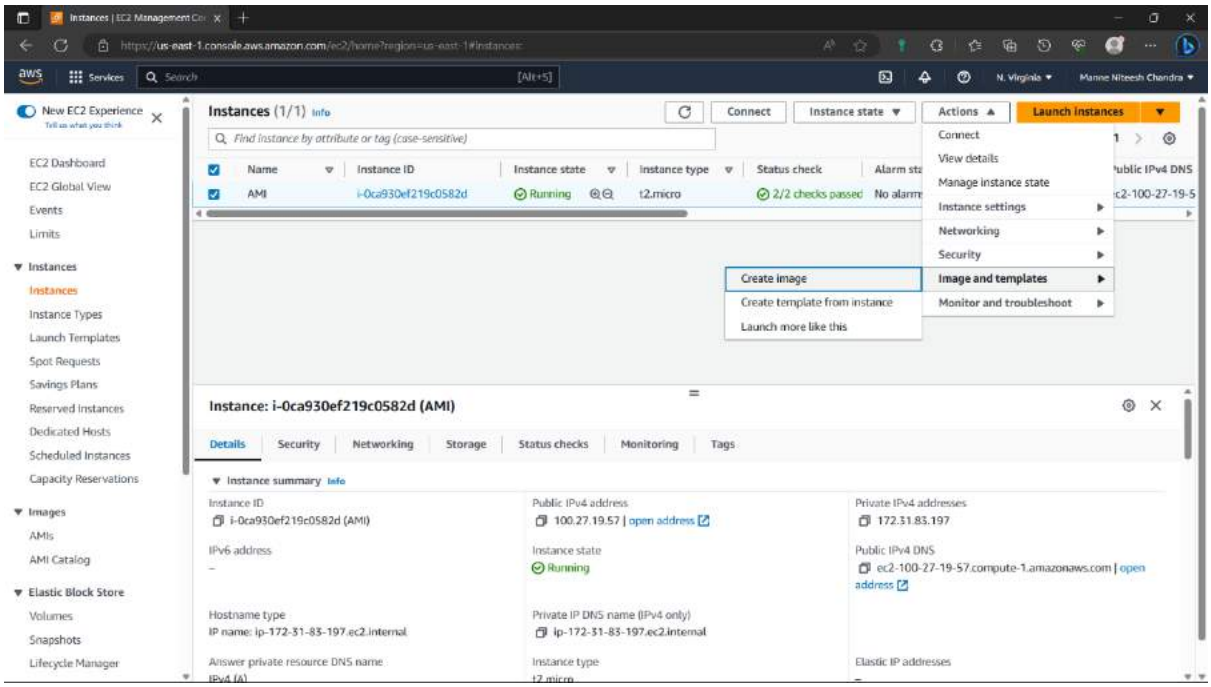
EBS Snapshot-It is used to share the data in different availability zones.

AMI:

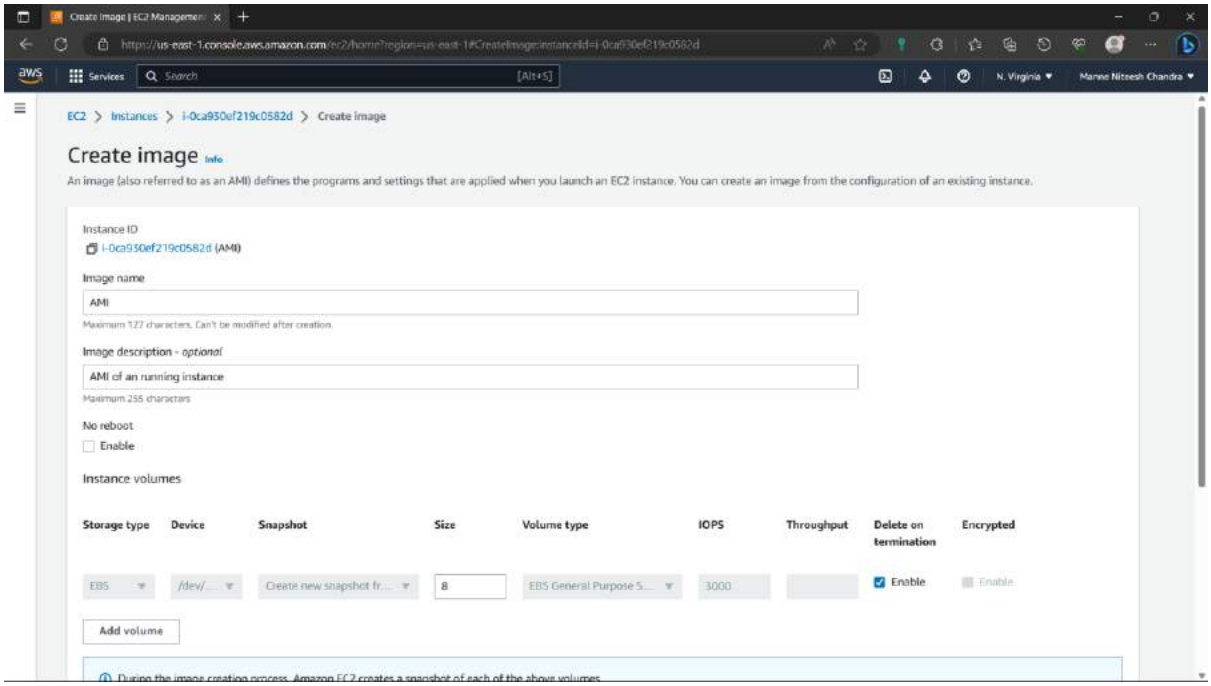
208)Create an EC2 instance by following the steps:-(129-139).



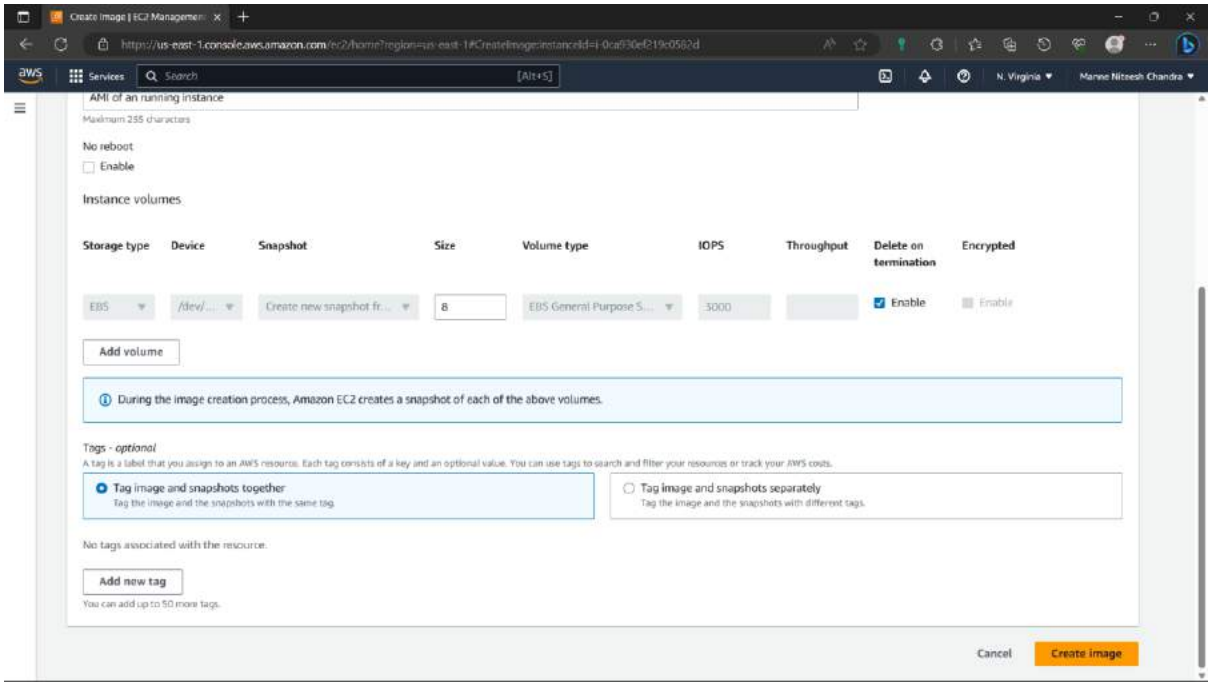
209)Select the instance and click on “actions”→”Images and Templates”→”Create image”.



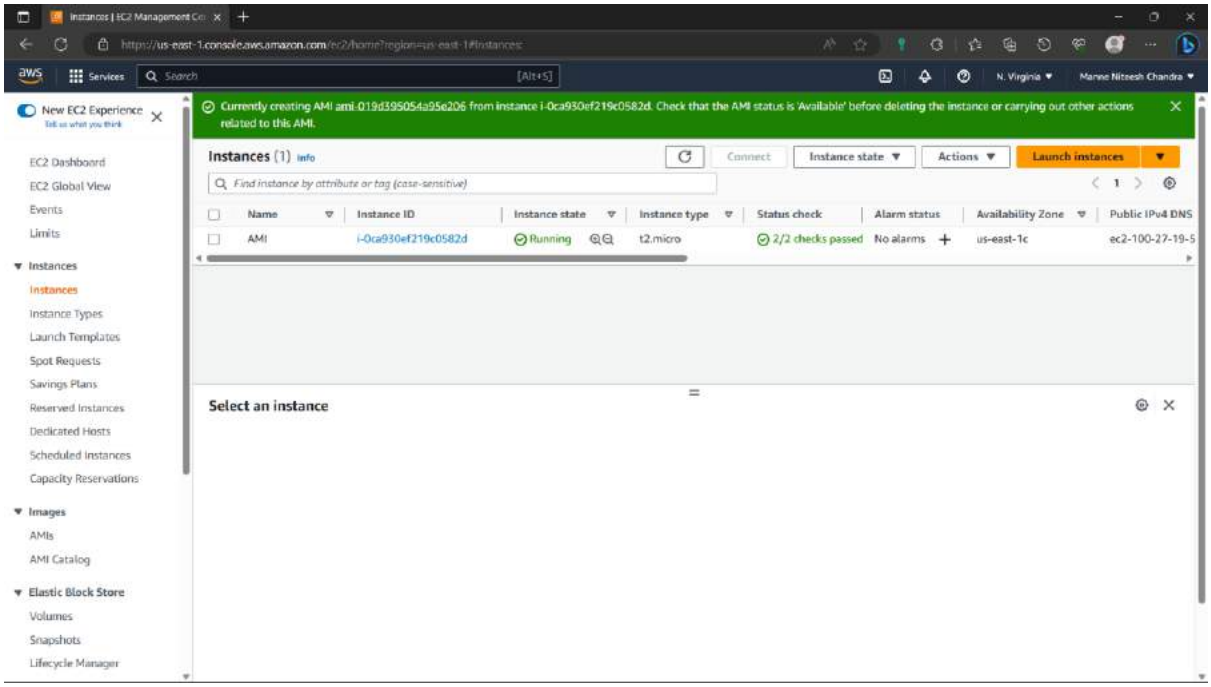
210)Give name, description of your choice.



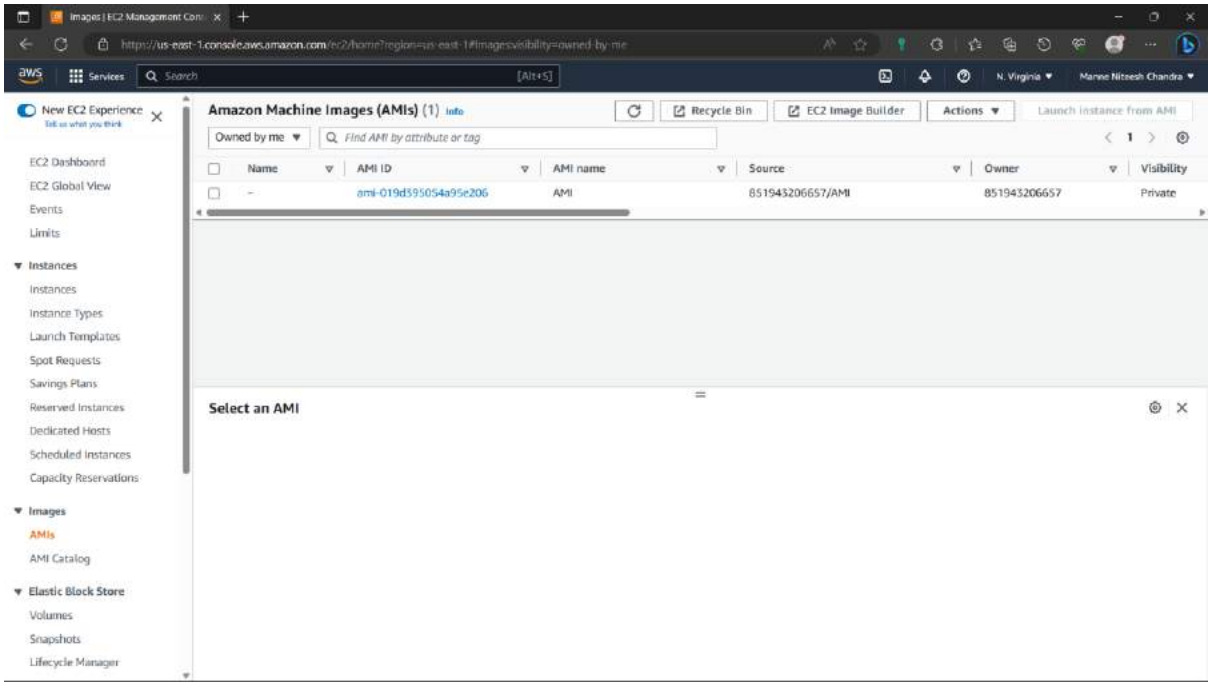
211)Click on “Create image”.



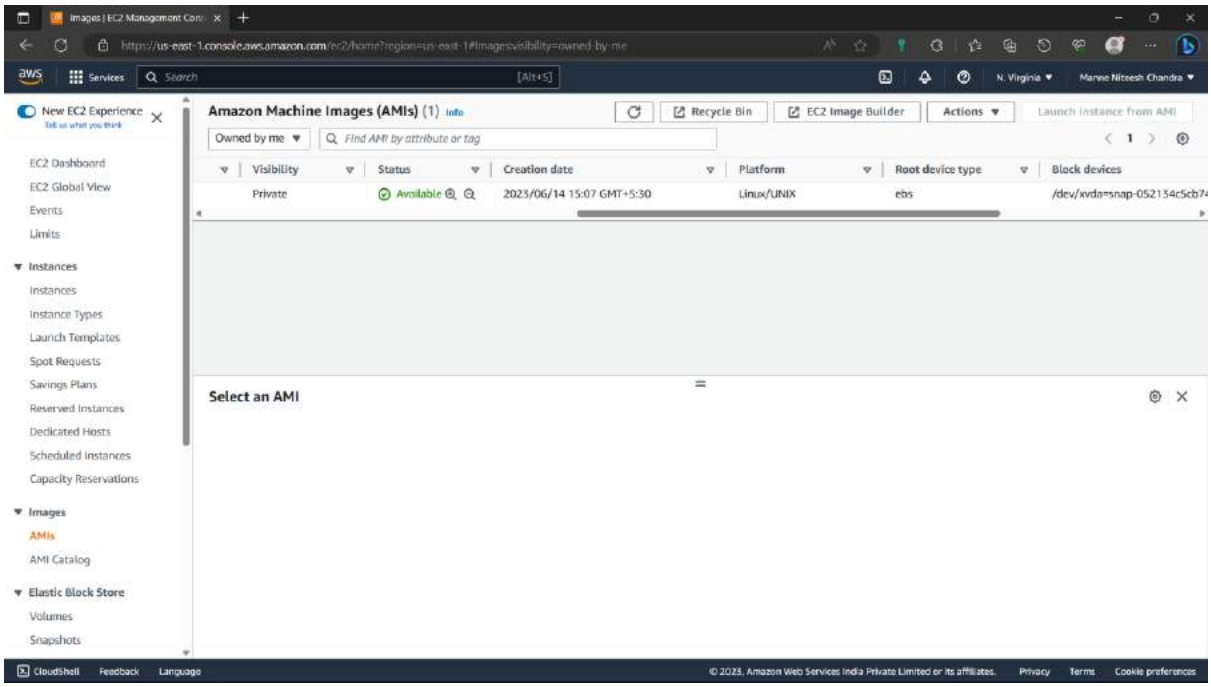
212)The image has been created.



213)You can see the created image in “AMIs” of Images at the navigation panel.

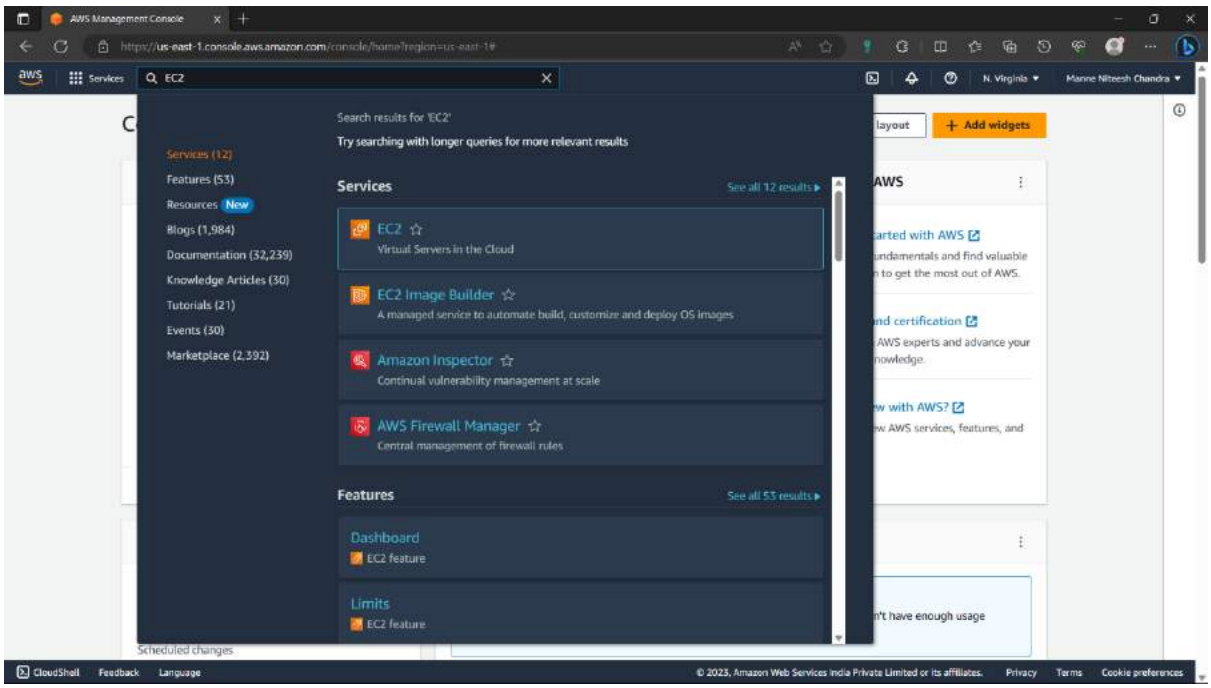


214)Wait till the “status” in AMI shows “Available”, then we can use these image.

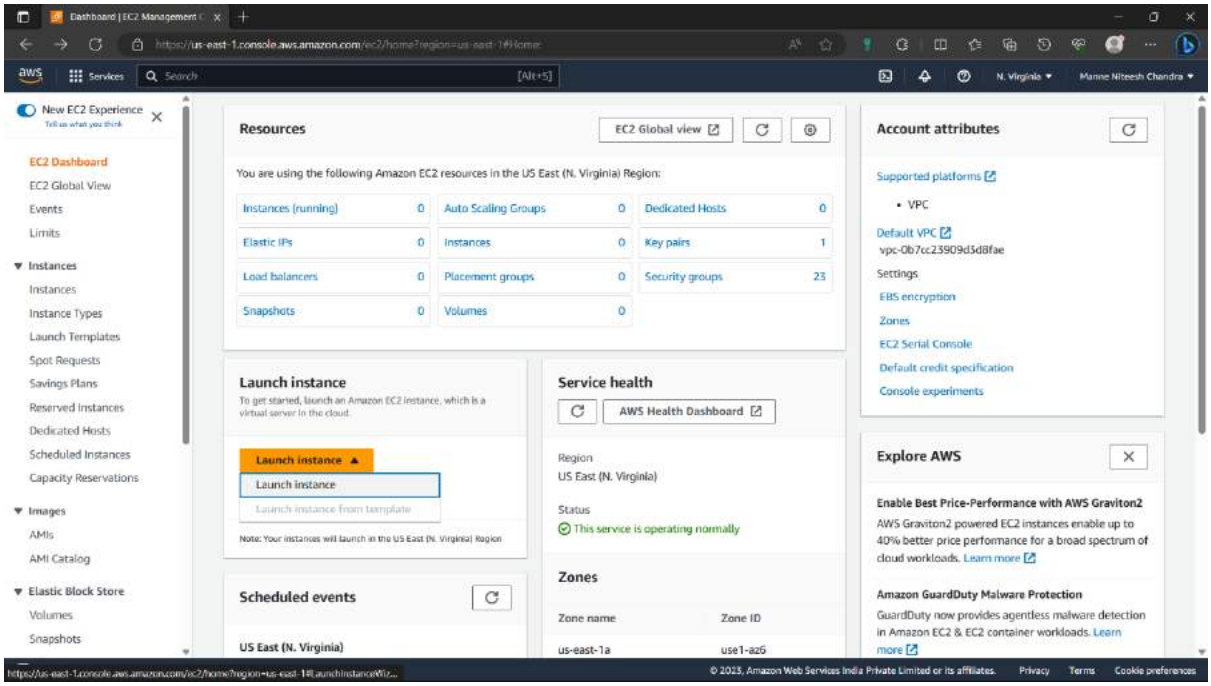


Load Balancers:

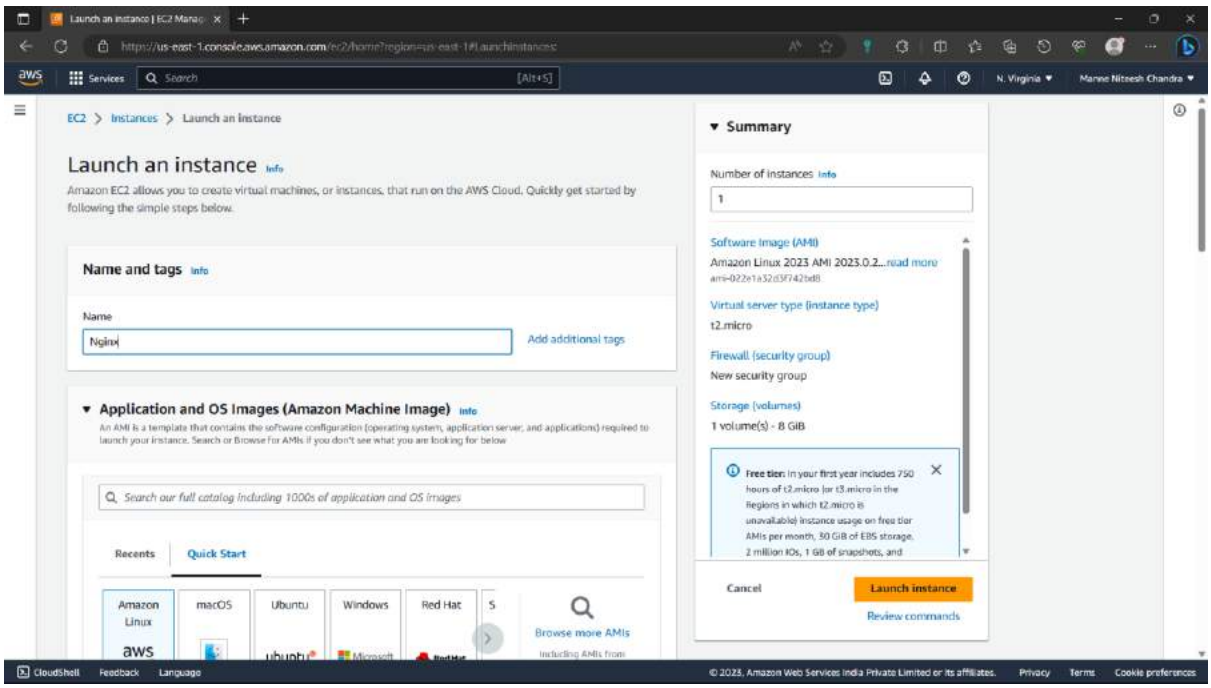
215)Type “EC2” in search bar and select the “EC2” option from below.



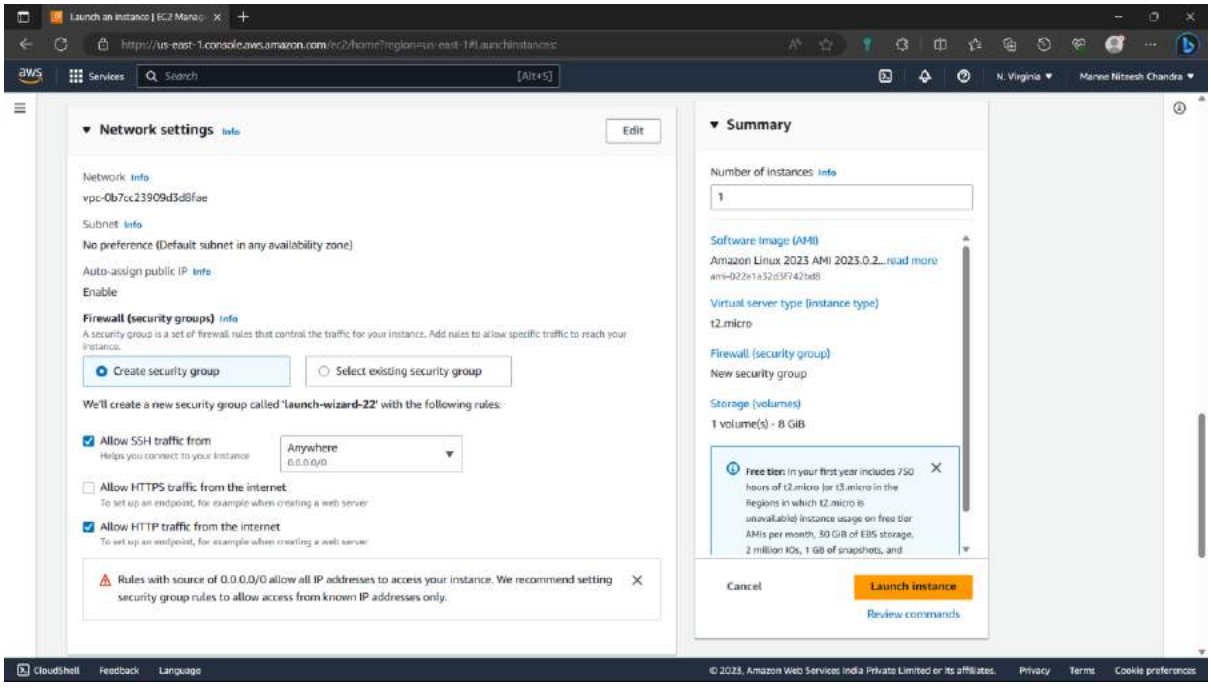
216)Select “launch instance” option to create a new instance.



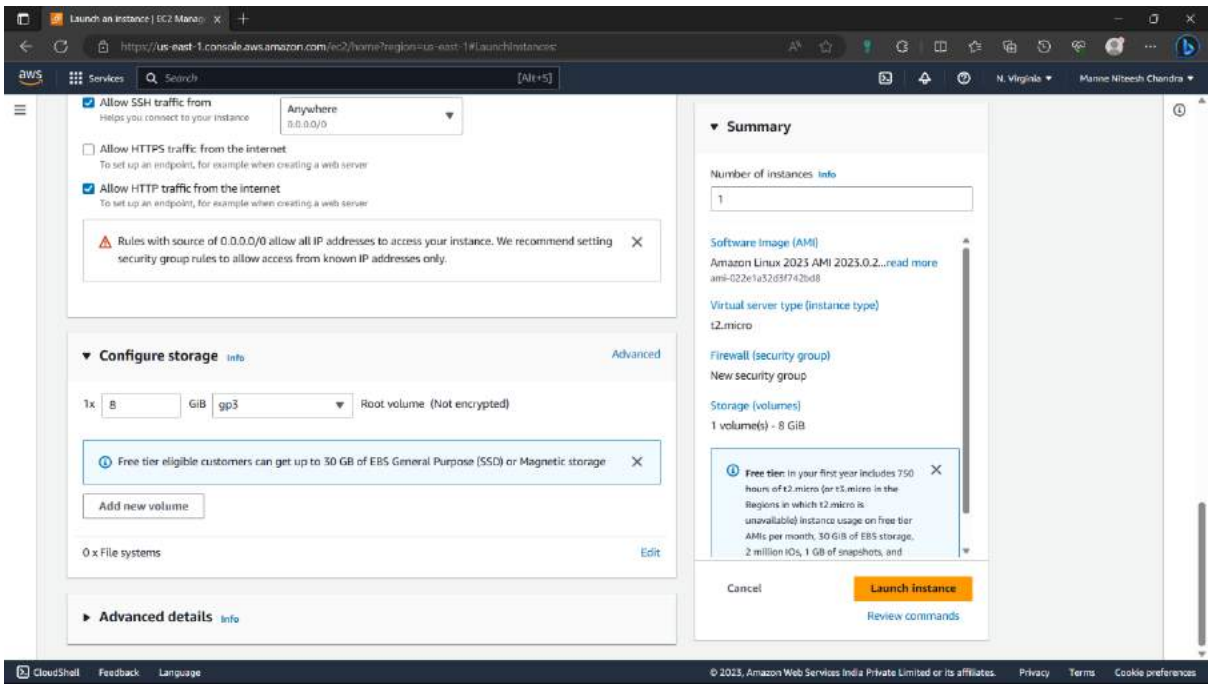
217)Give name “nginx” let the AMI be default.



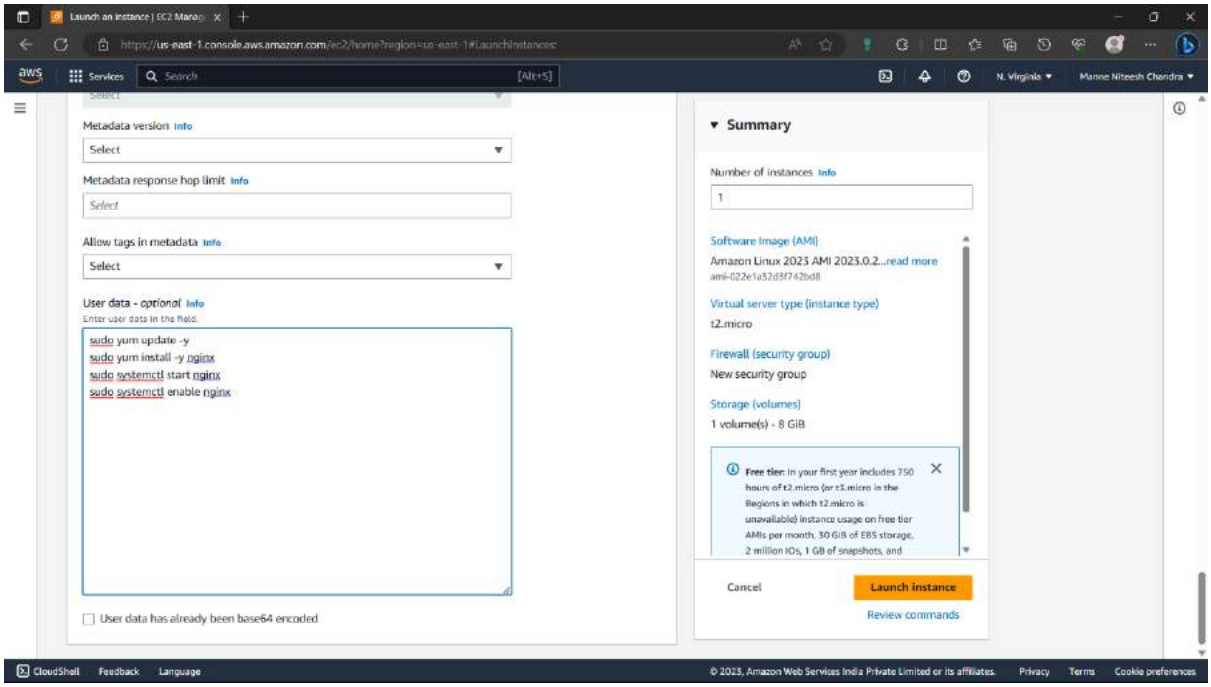
218)Enable the “HTTPS Traffic” in “network settings”.



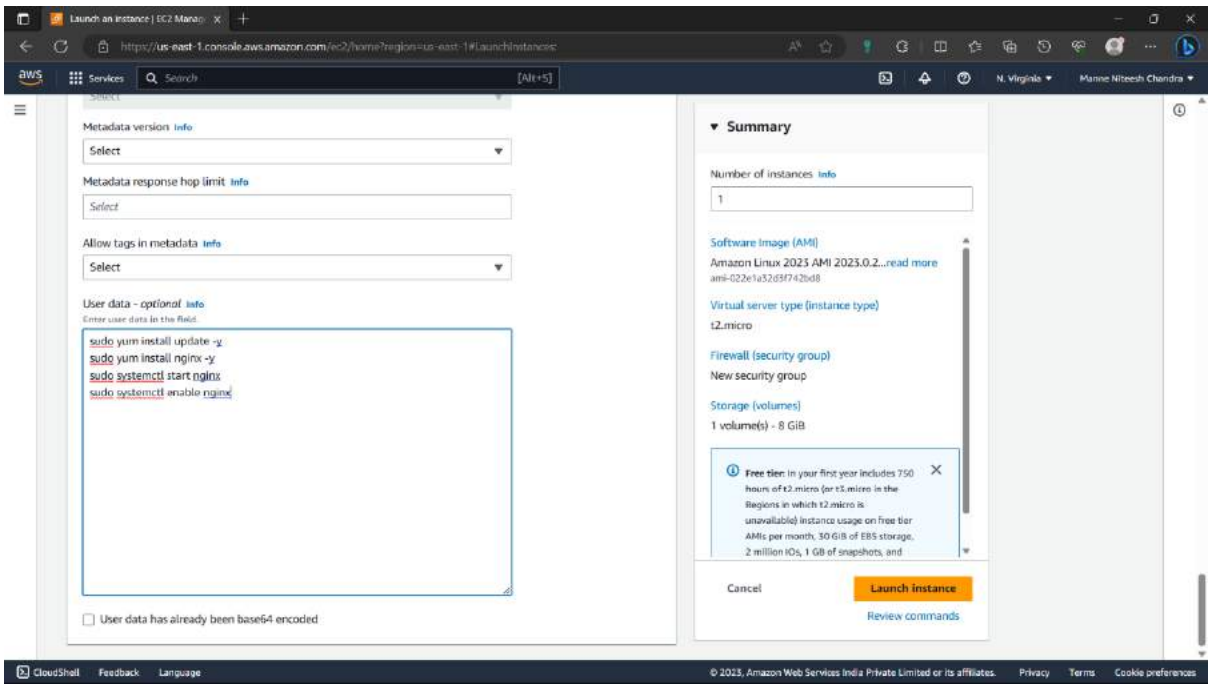
219)Click on “Advanced details” option.



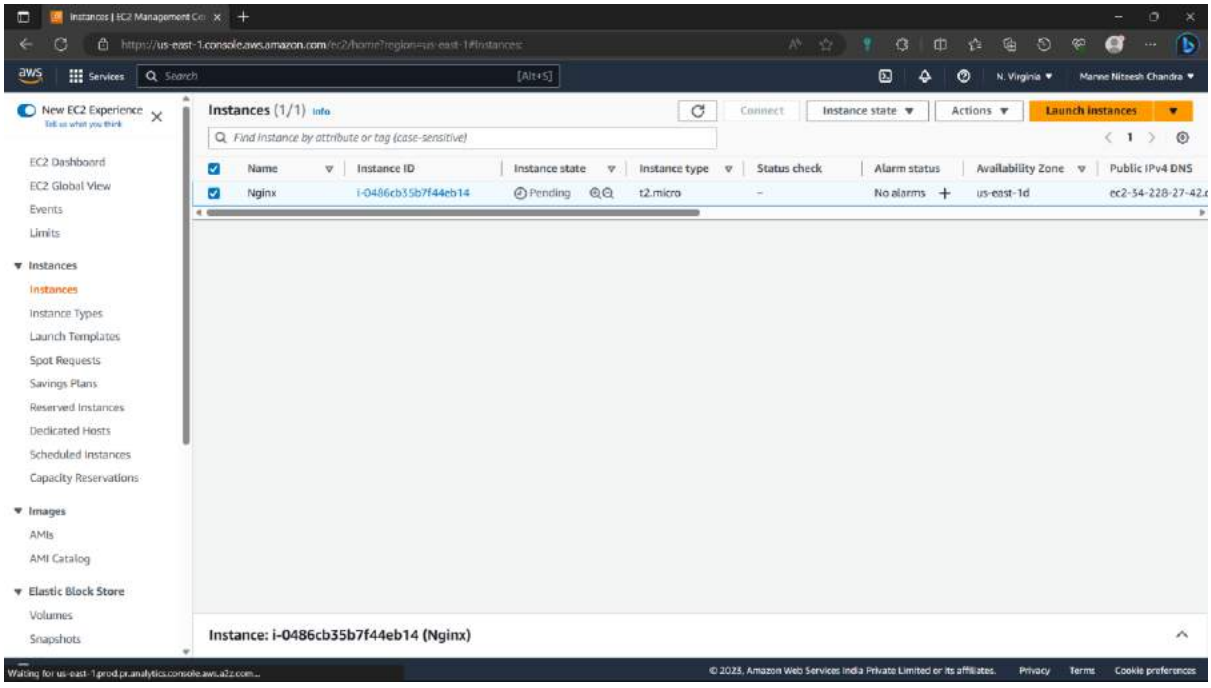
220)Type below commands in image in “user data” box at bottom of the page.



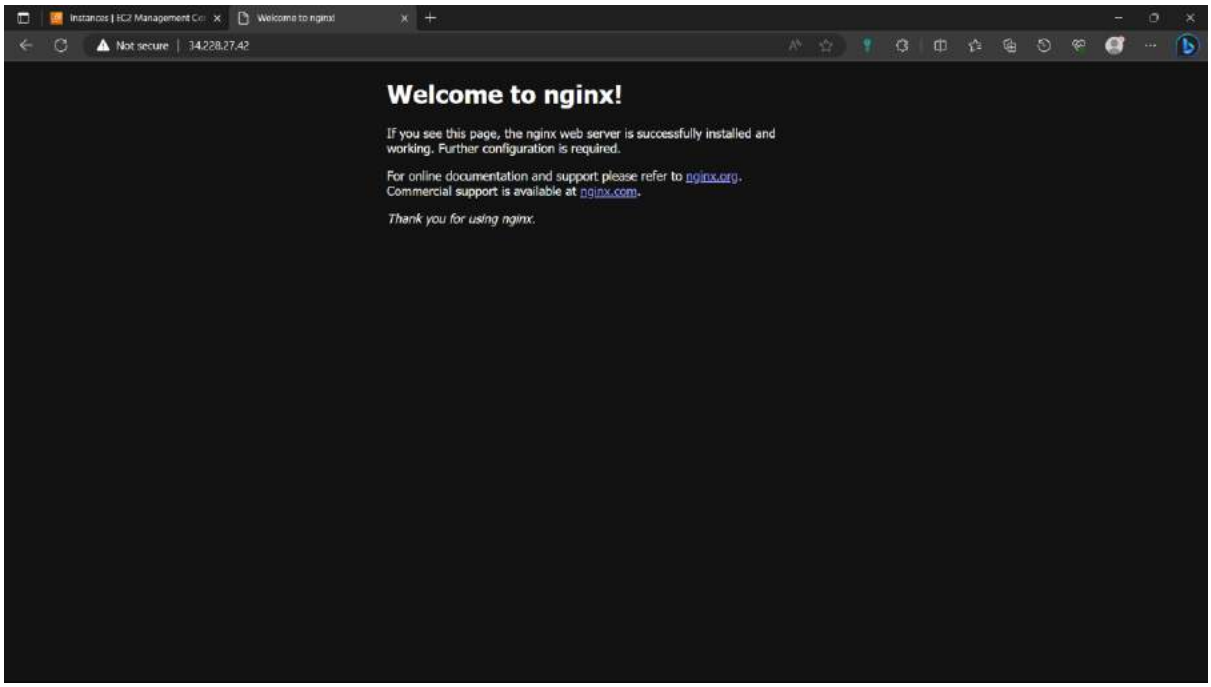
221)Click on “Launch instance” to create the instance.



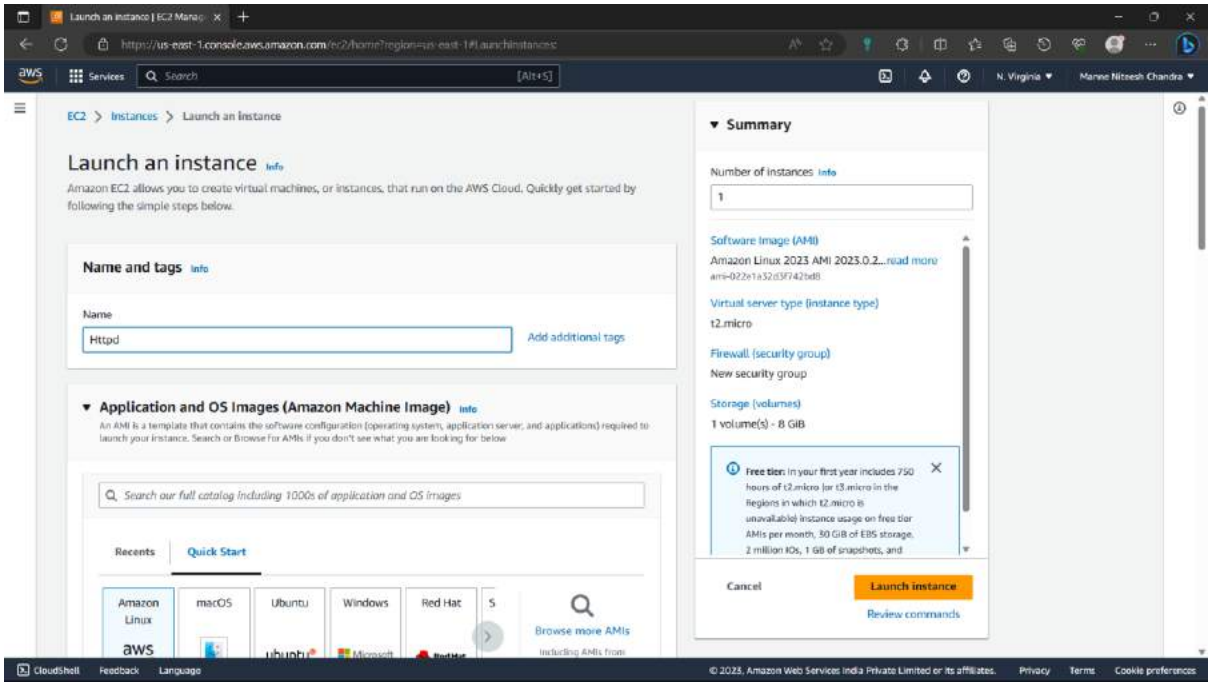
222)Now instance is created.



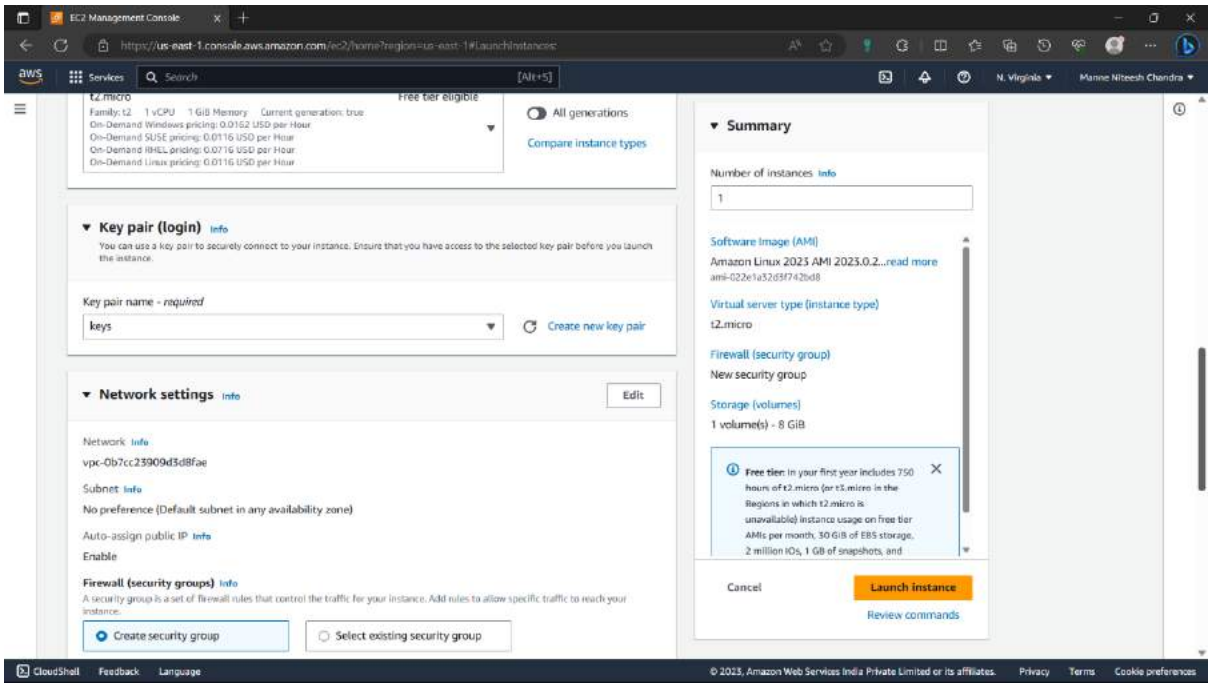
223)Select the instance you can see the details below, copy the Ipv4 address of the instance and past it in a new browser.



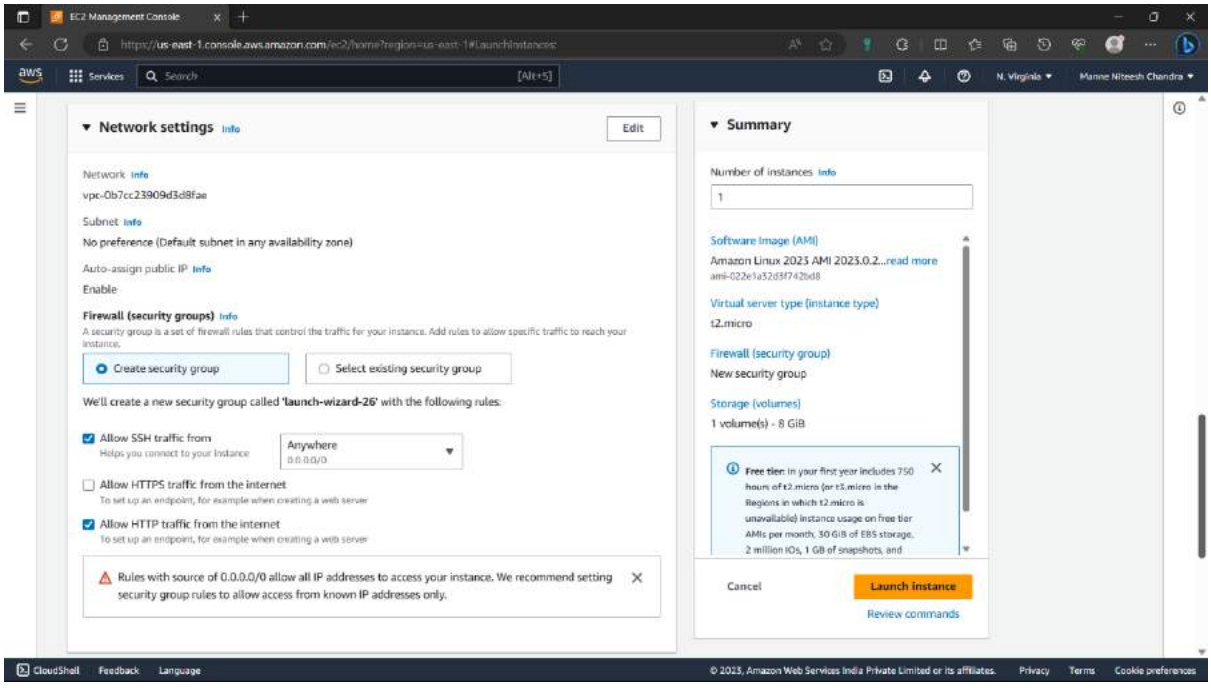
224)Now create a new instance for “Apache Httpd” , Click on “Launch instance” option. Give “Httpd” as name and let the AMI be default.



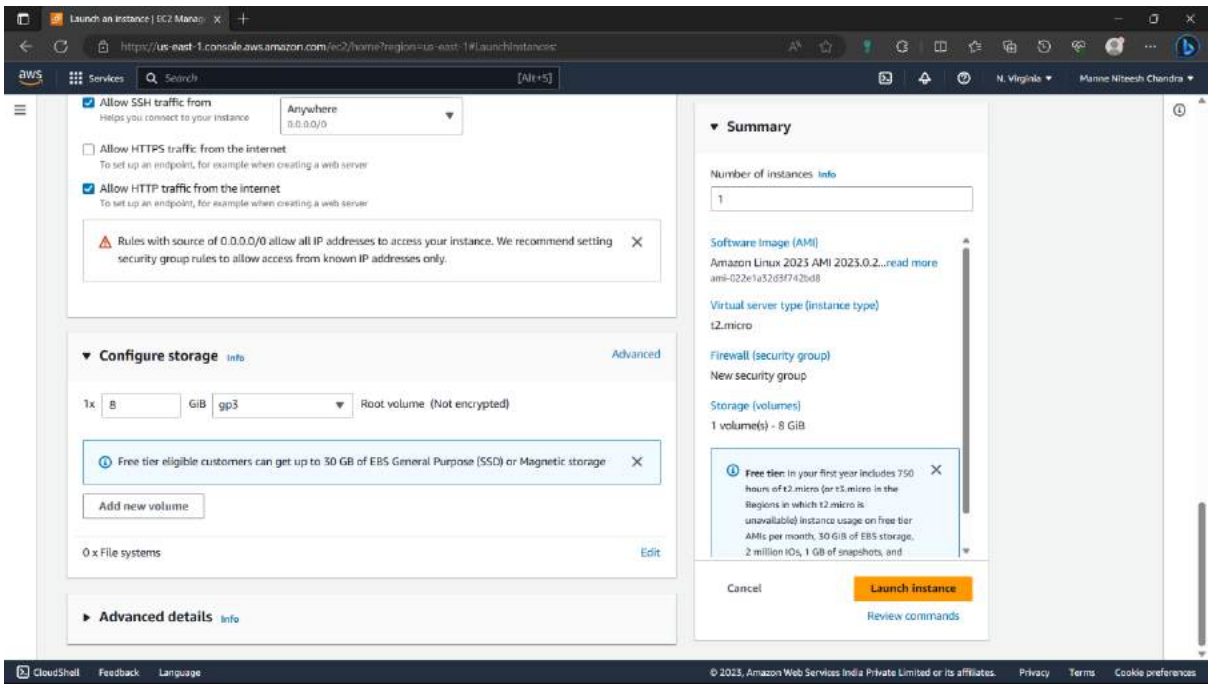
225)Select the key pair you have create new a pair key pair by clicking on “create a new key pair option”.



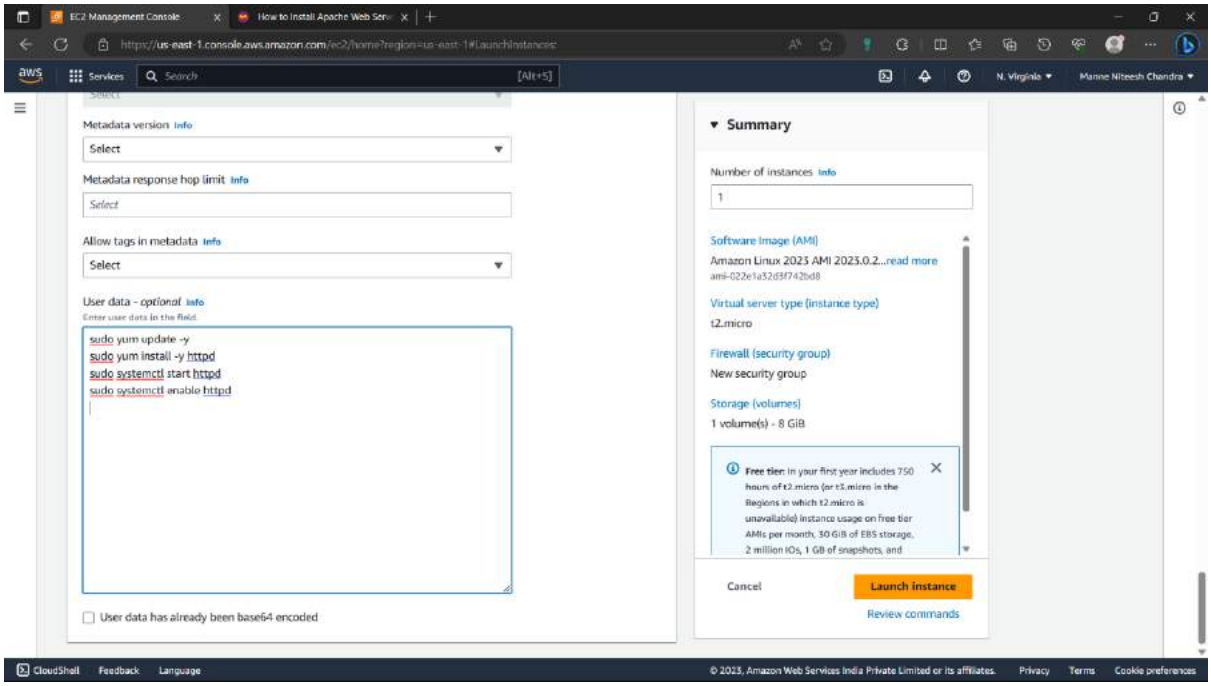
226)Enable the “Https traffic” option form “network settings”.



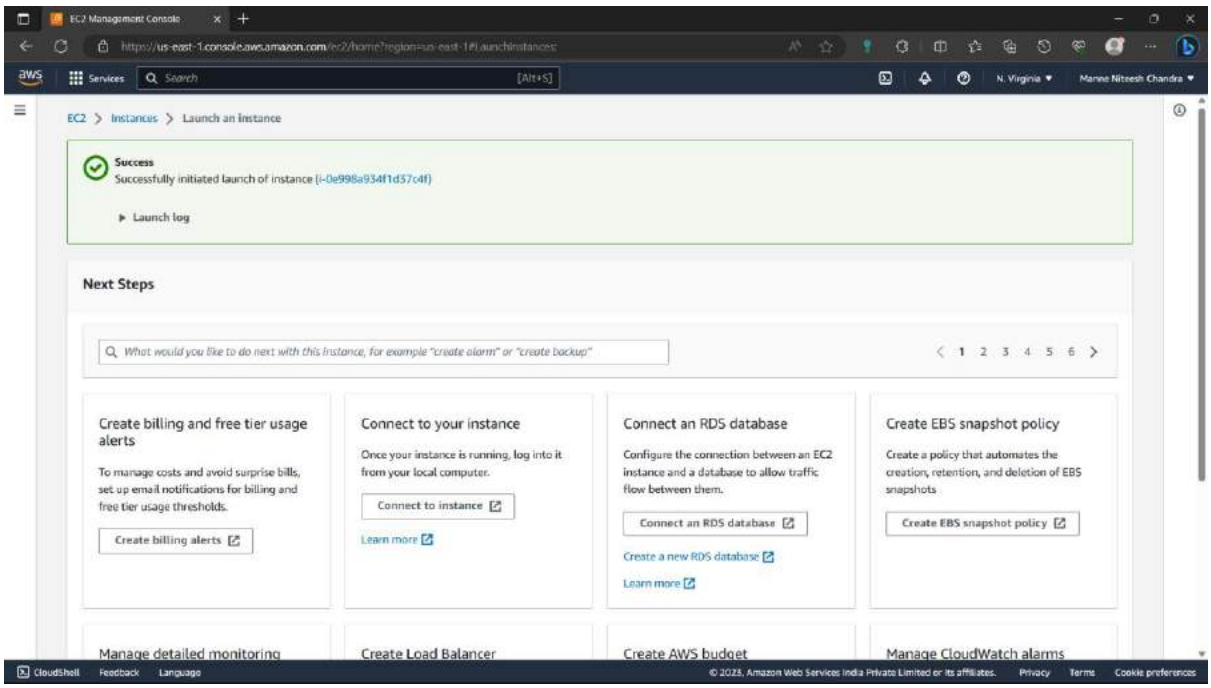
227)Click on “Advanced details” option below.



228) Type below commands in image in “user data” box at bottom of the page.

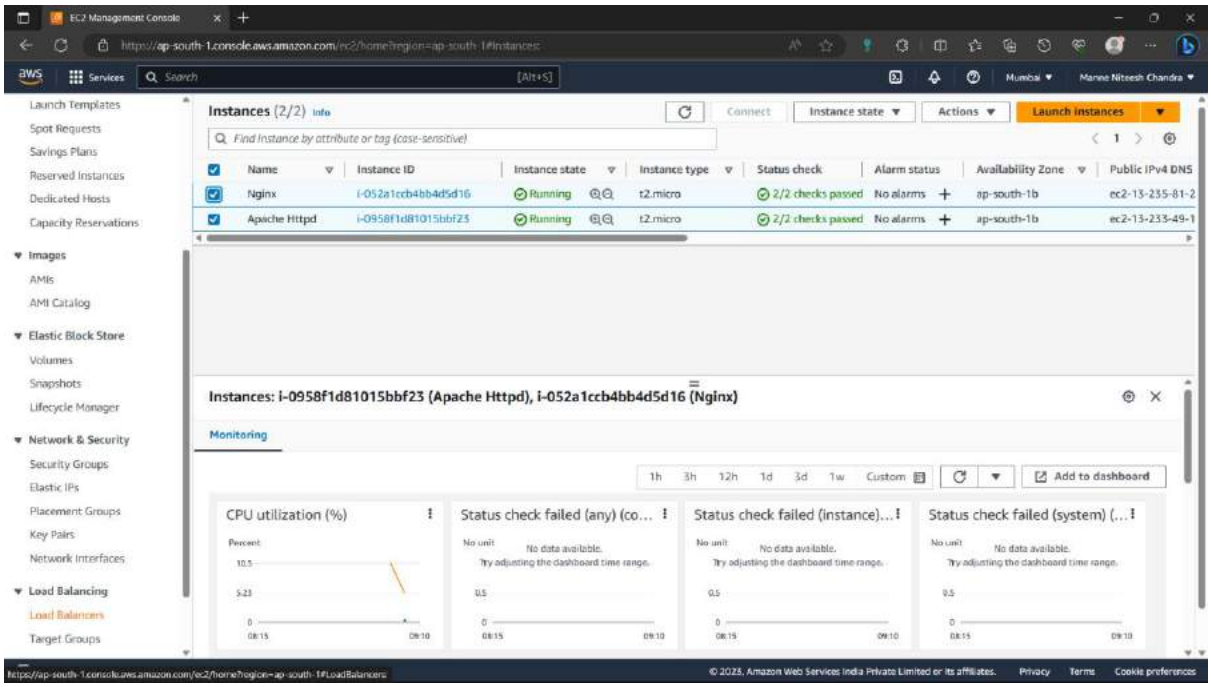


229)Now the instance is created.

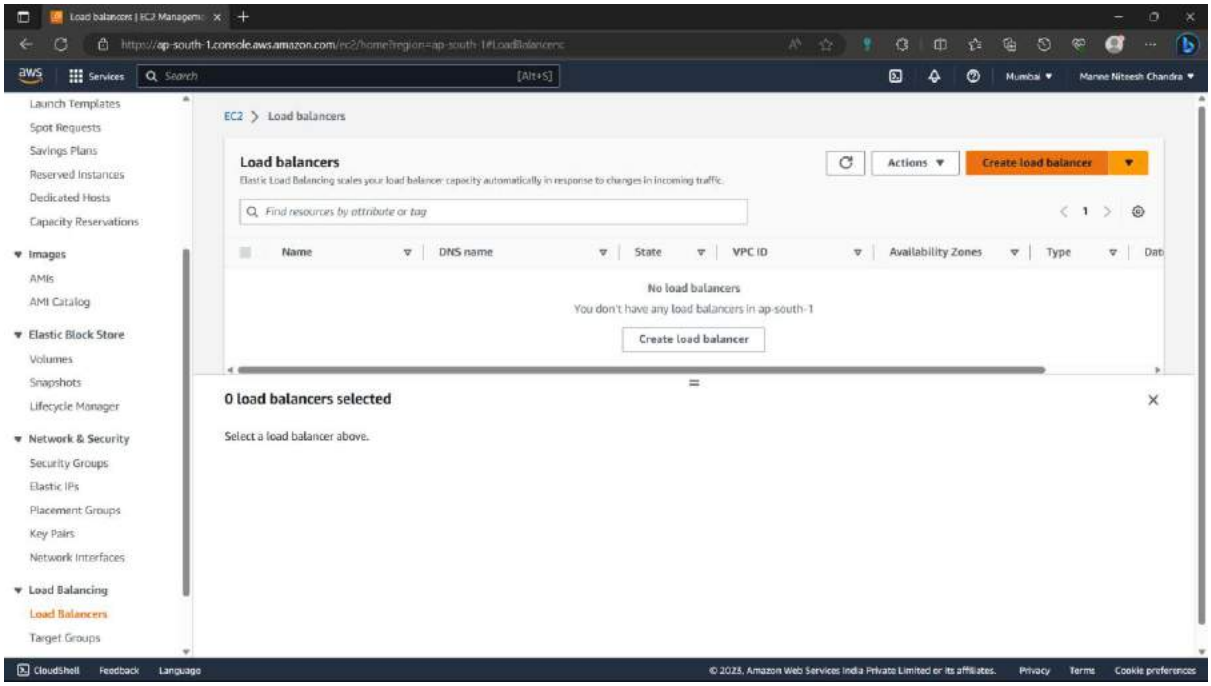


230) Select the instance you can see the details below, copy the Ipv4 address of the instance and past it in a new browser.

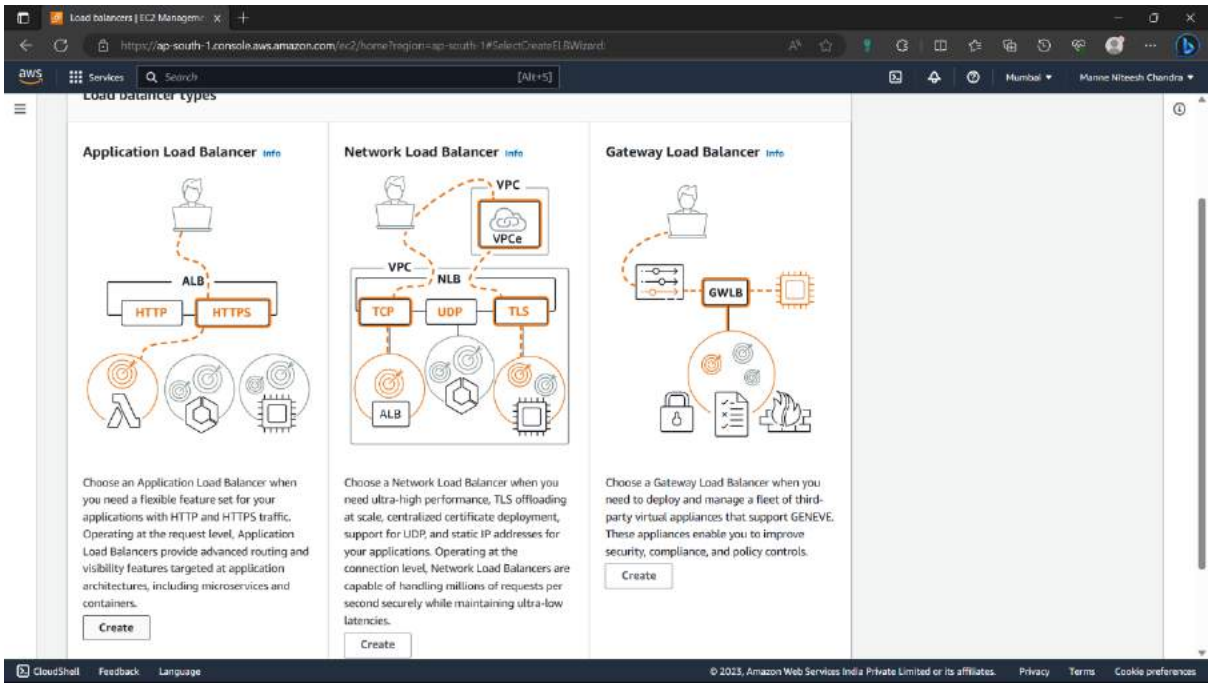
231)Click on “load balancer” in “load balancing” in below the left navigation pane.



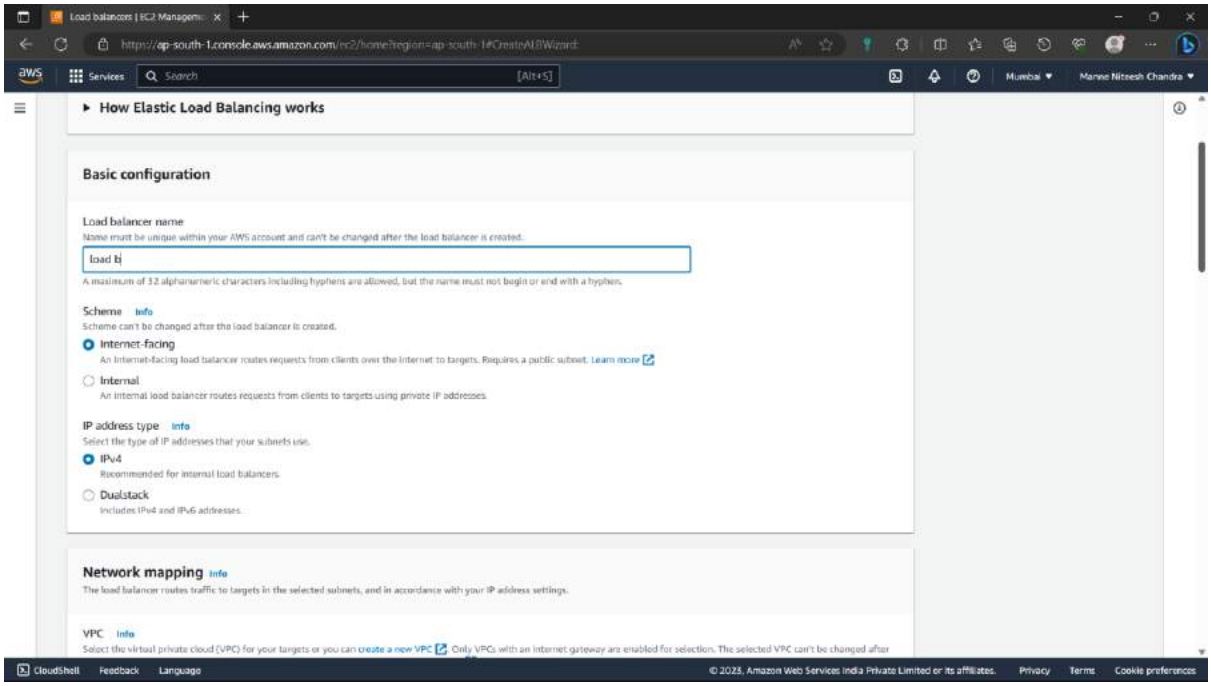
232)Click on “create load balancer” option to create load balancer.



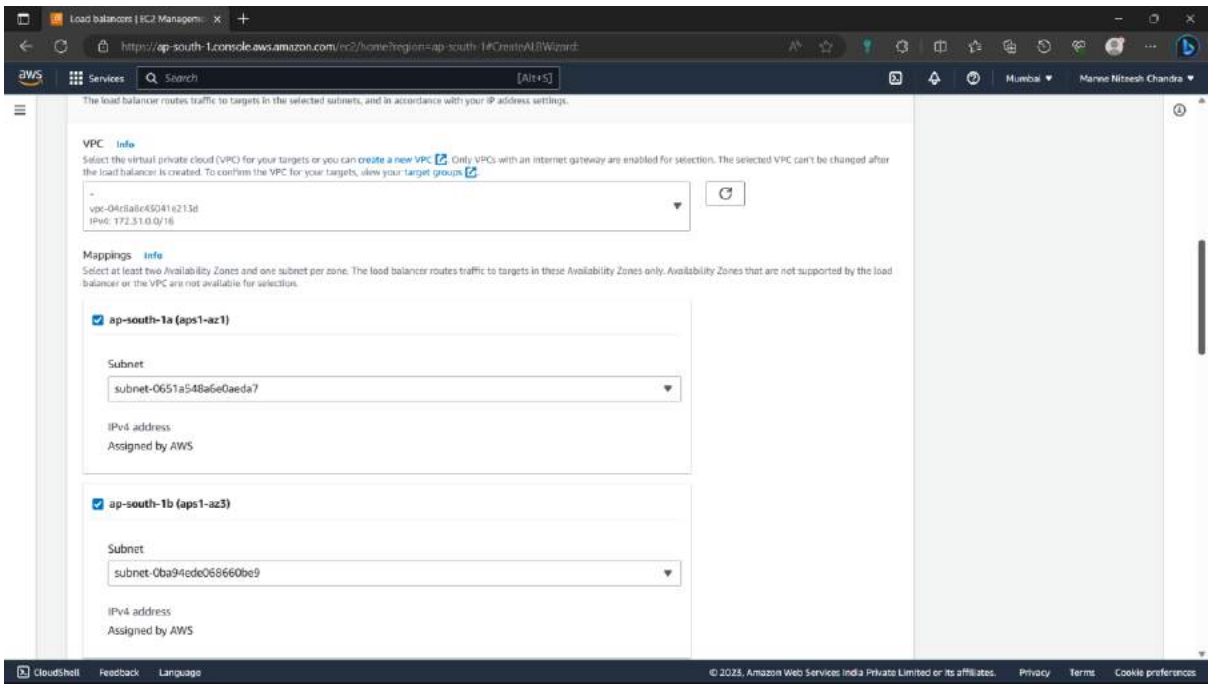
233)Select “Application Load Balancer” because we are creating both the EC2’s under same VPC.



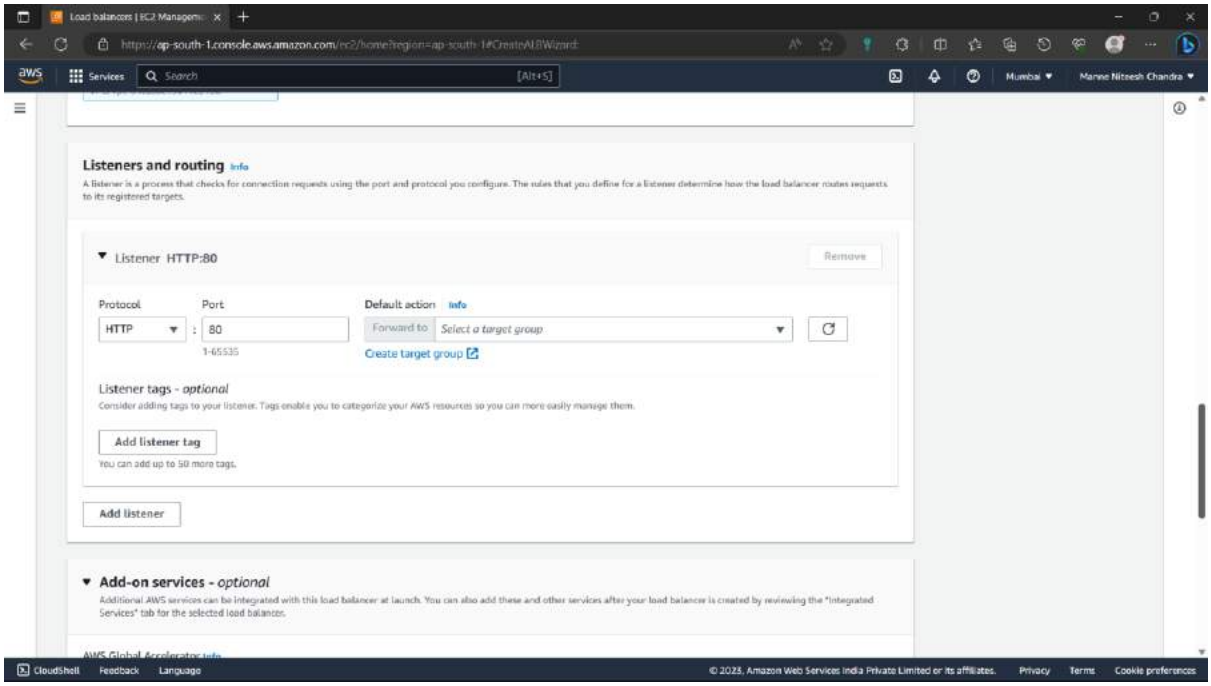
234)Give a name to the load balancer and let everything be default.



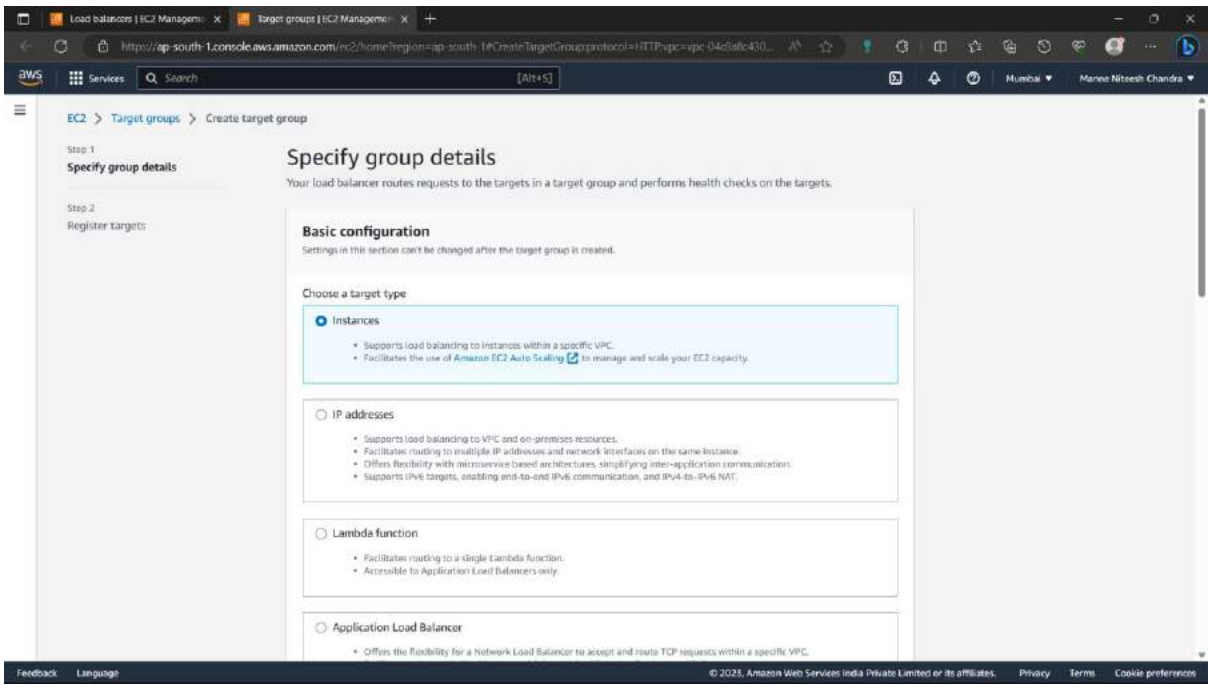
235)Select VPC, 2 Availability zones for subnets.



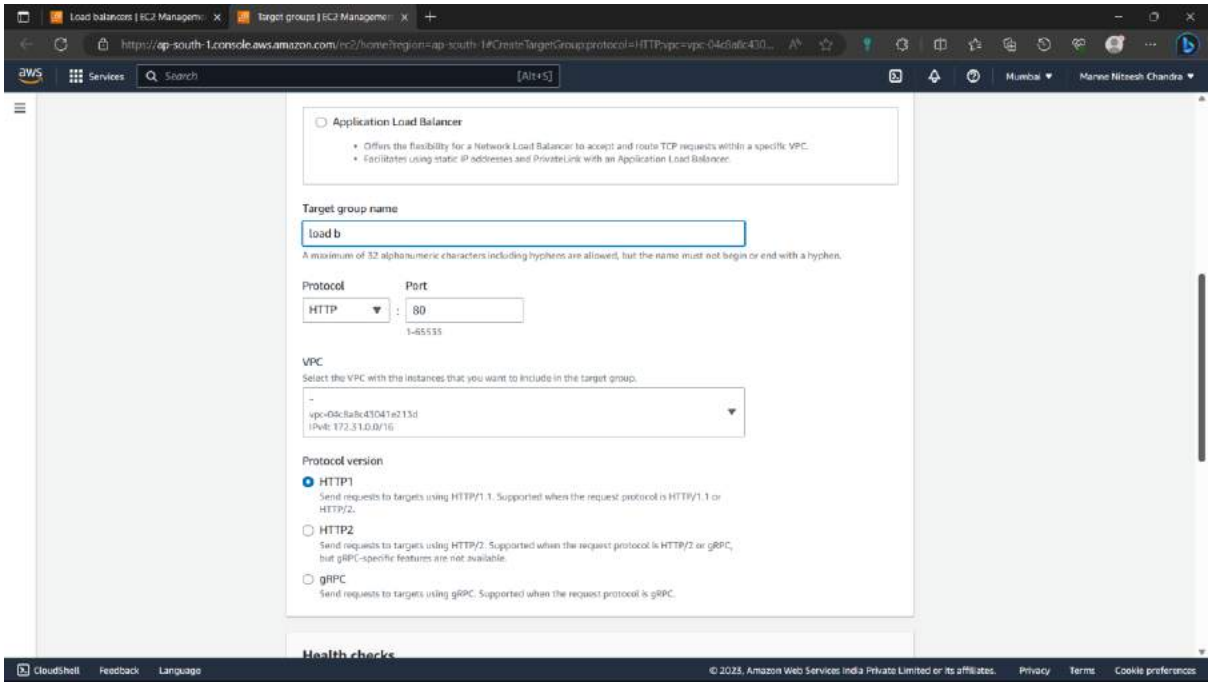
236)Port number “80” click on “create target group” below the bar in blue color.



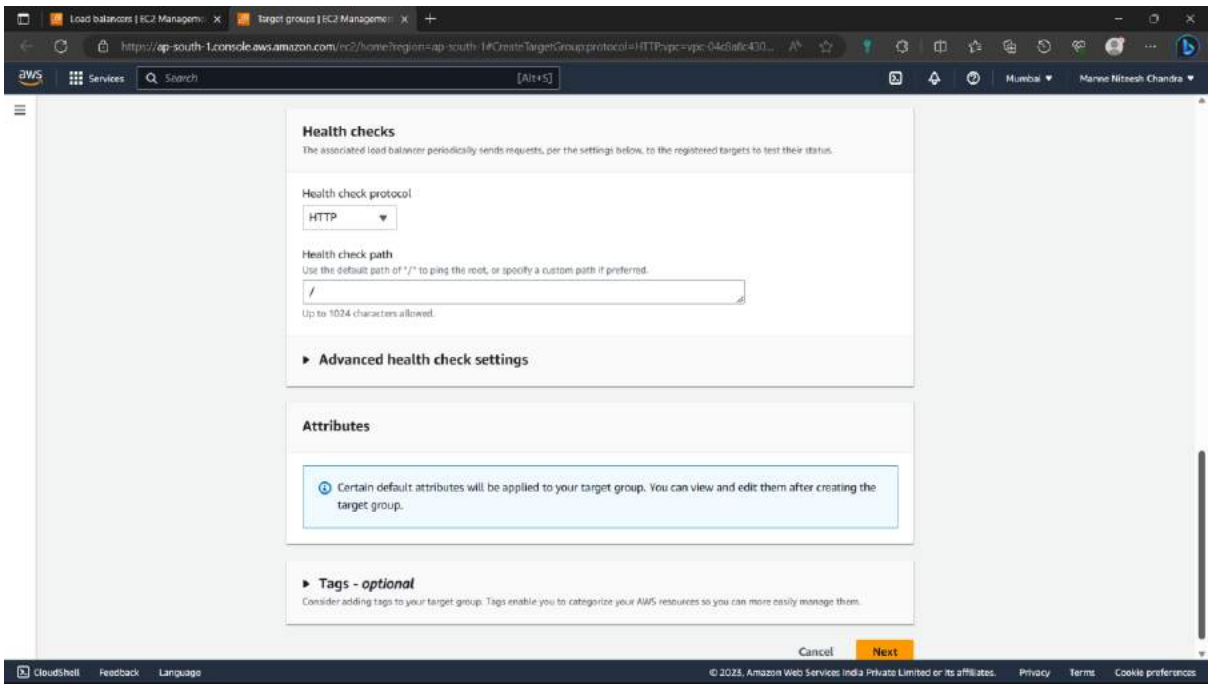
237)Now the target group is opened in new tab and select the “instance” option.



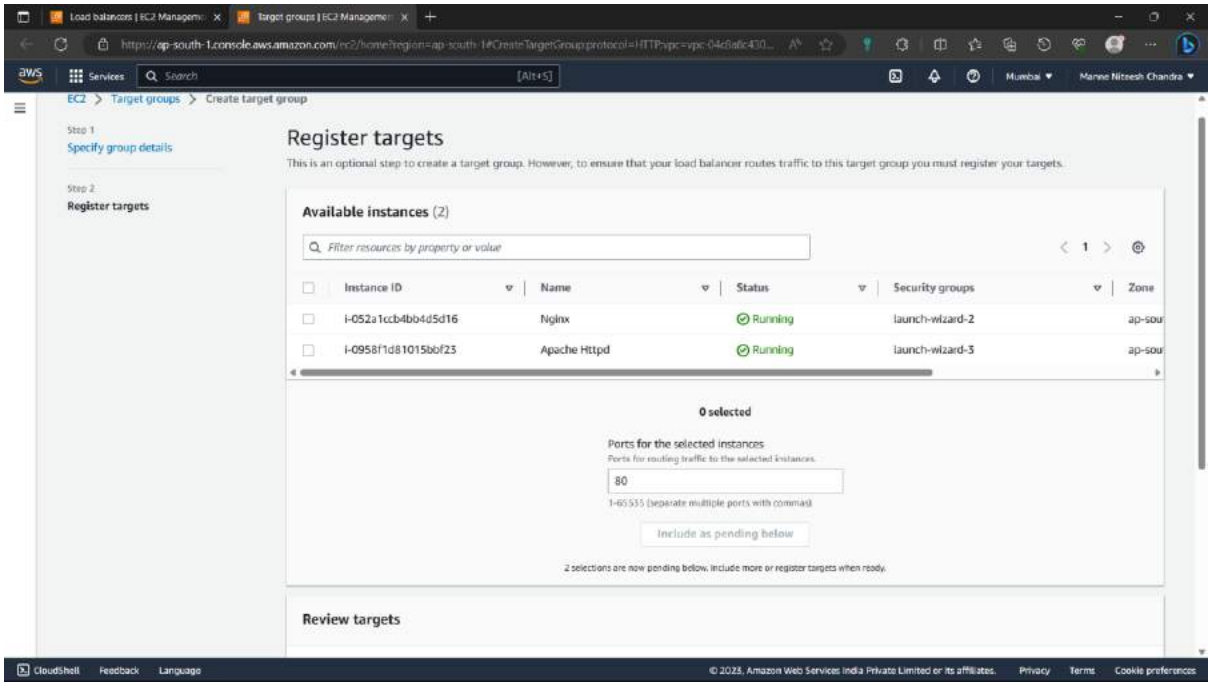
238)Give a name to the target group and let everything be default.



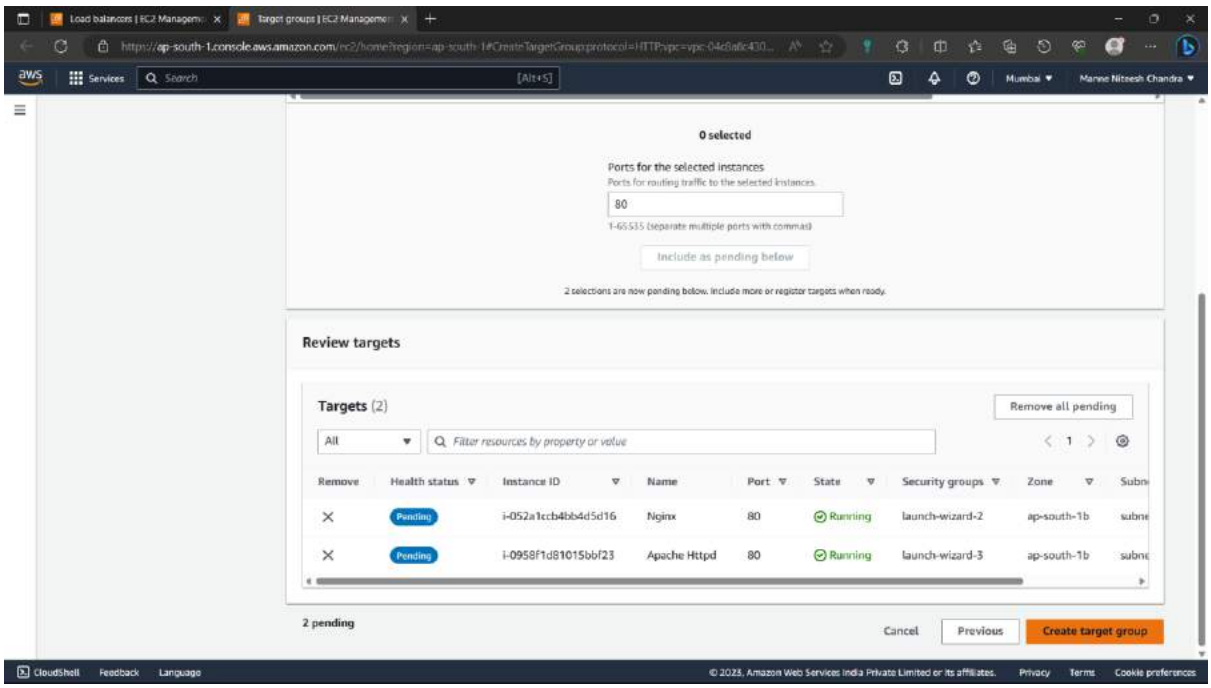
239)Click on “next”.



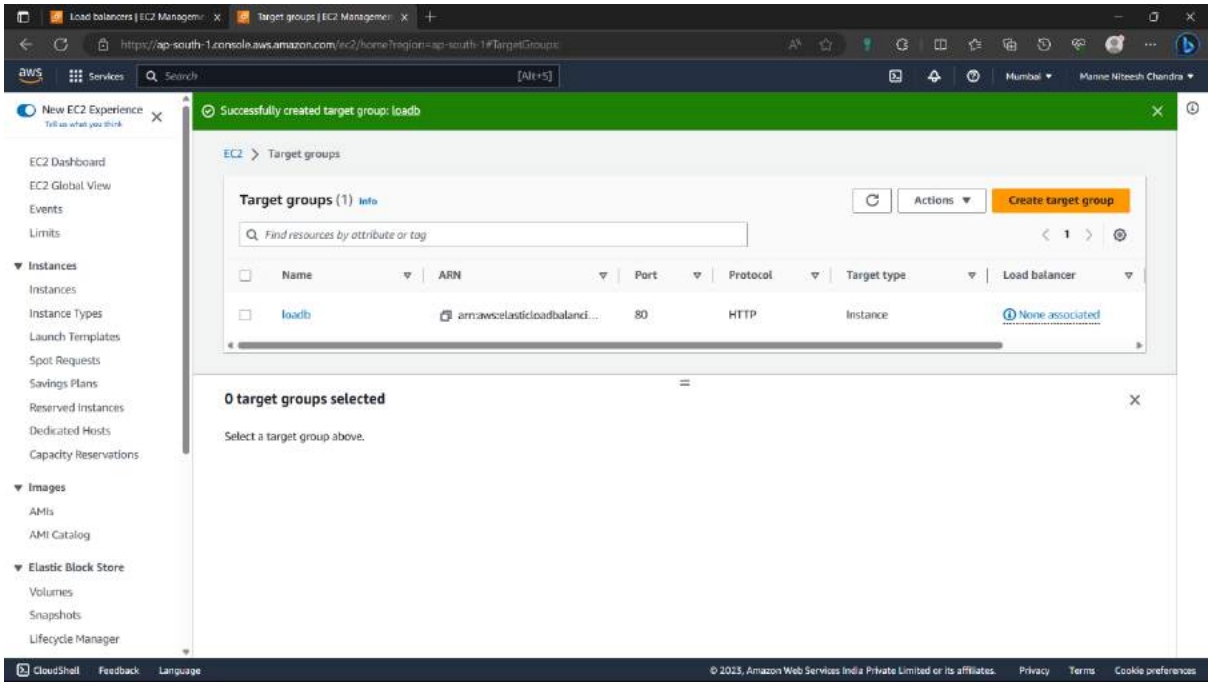
240)Select both the instances and click on “Including pending as below”.



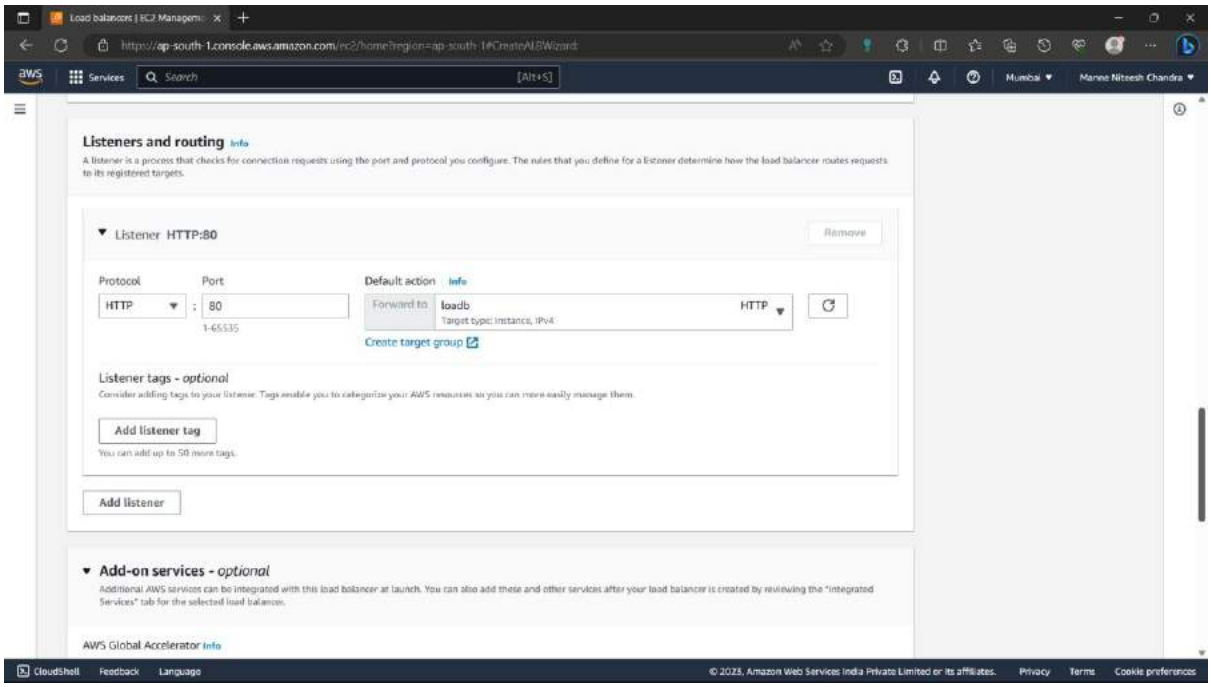
241)Click on “Create target group”.



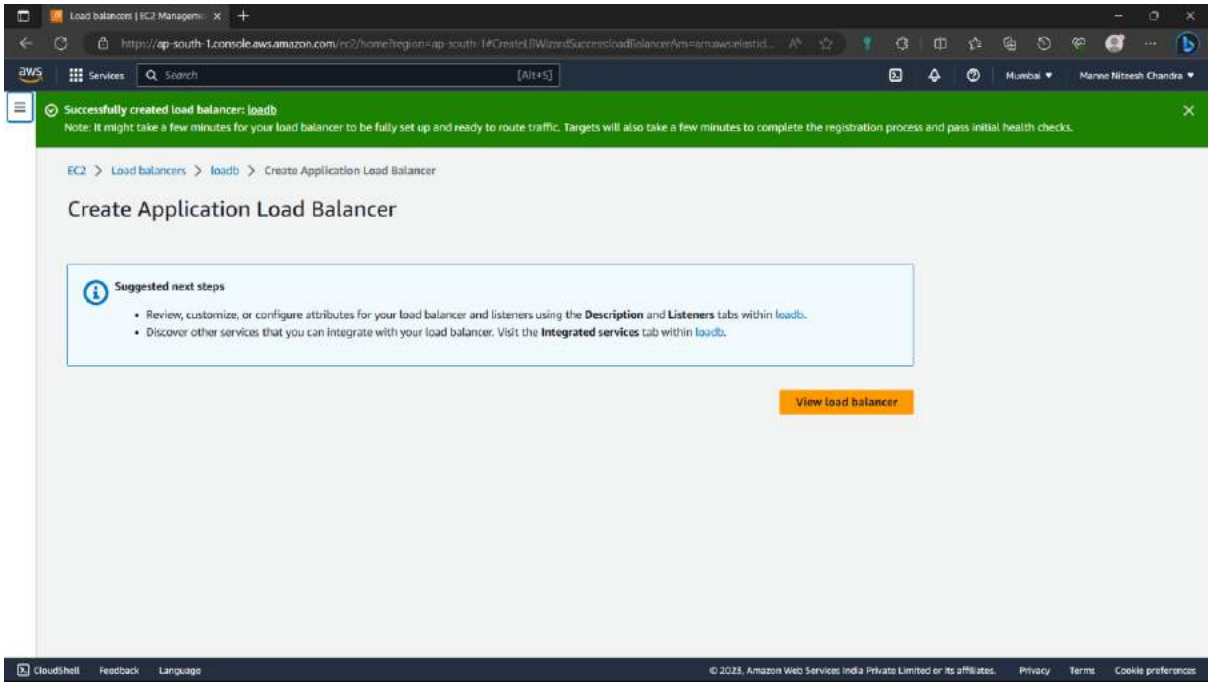
242)Now the target group is created.



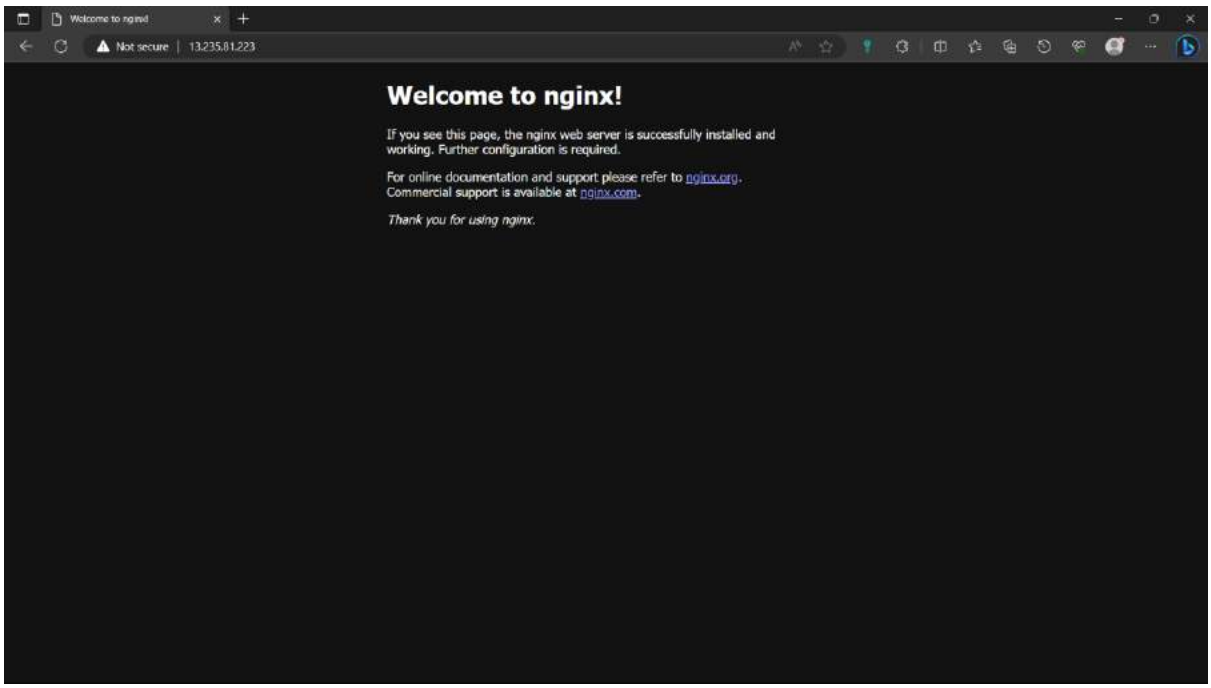
243)Go to the load balancer page and click on the refresh below which is beside the bar and select the created target group.



244)Click on “create load balancer”, The load balancer is created and assigned to the instances once refresh the load balancer and open the web pages of the instances by copying their Ipv4 addresses.

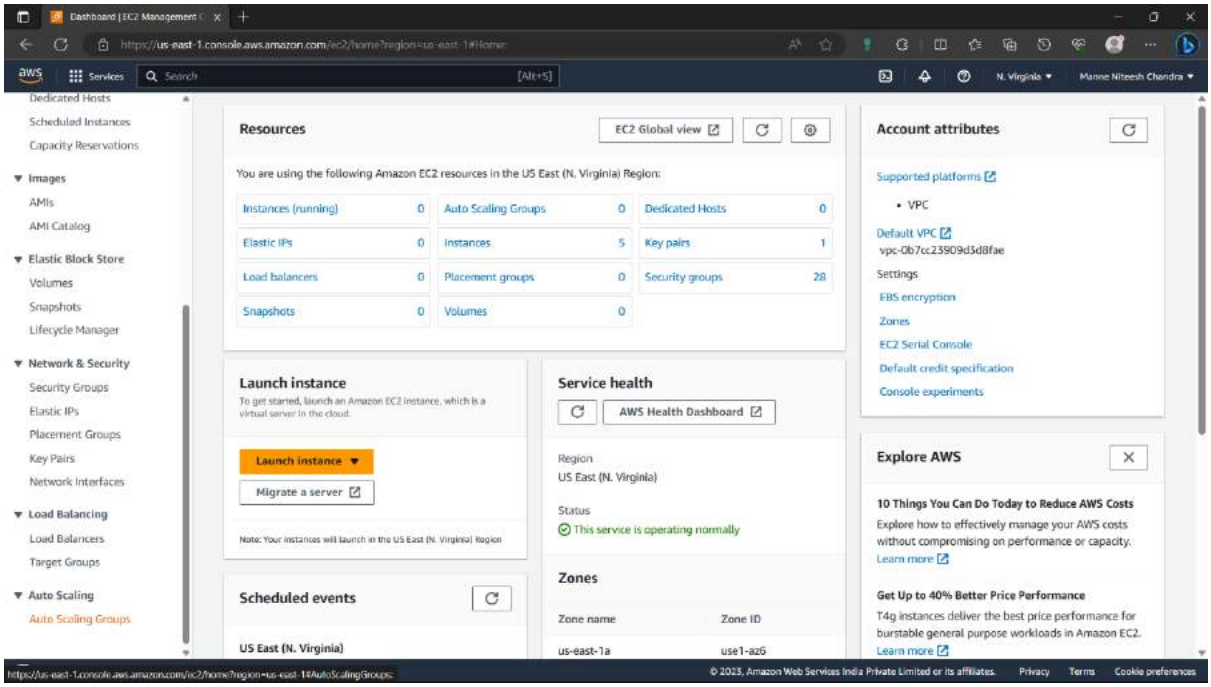


245)Now the webpages are opened.

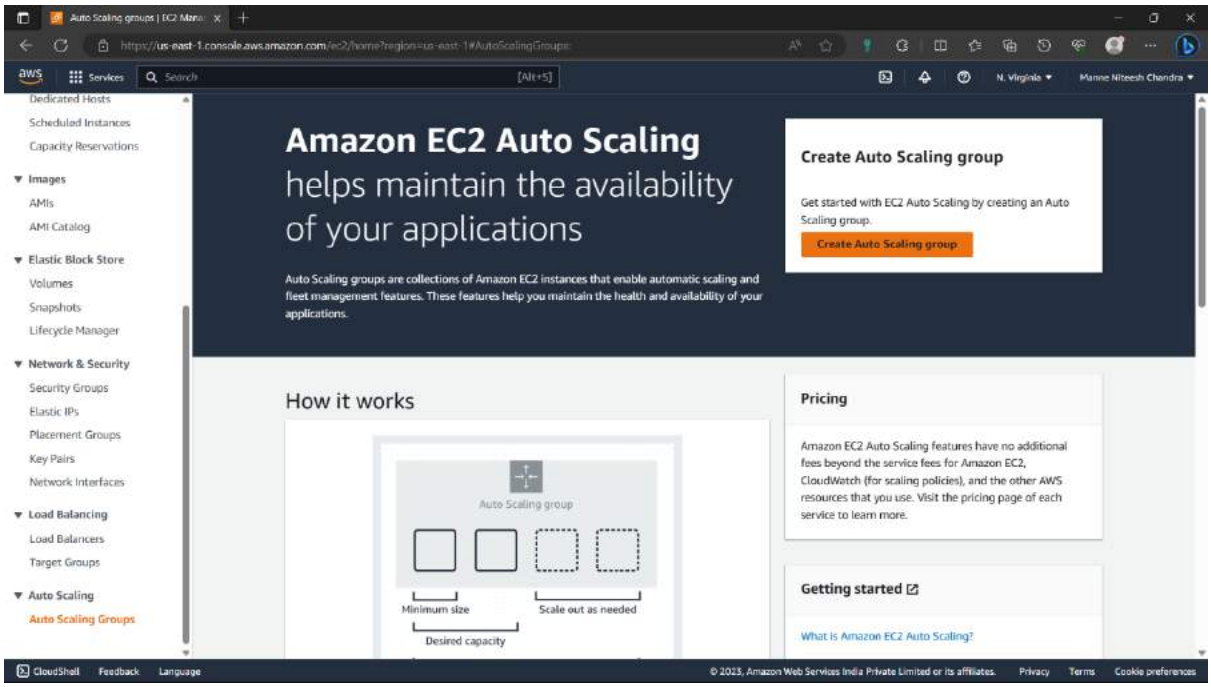


ASL and LG:

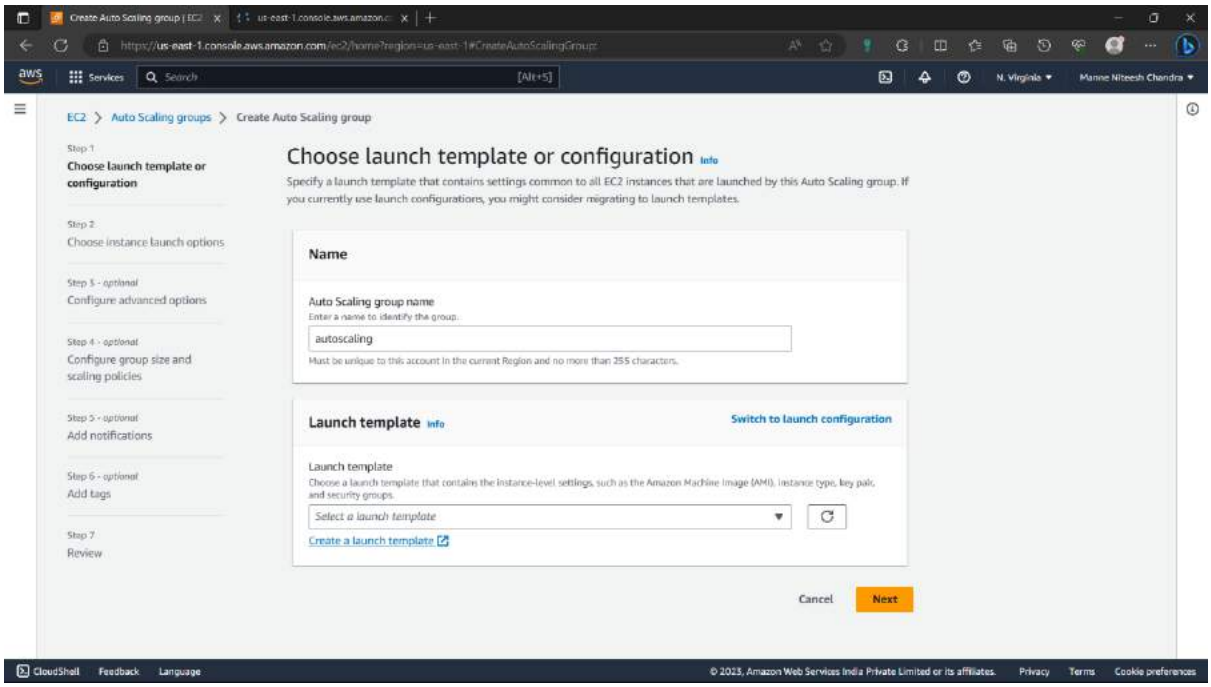
246)Click on “Auto Scaling Group” option “Auto Scaling” which is below the navigation panel of EC2.\



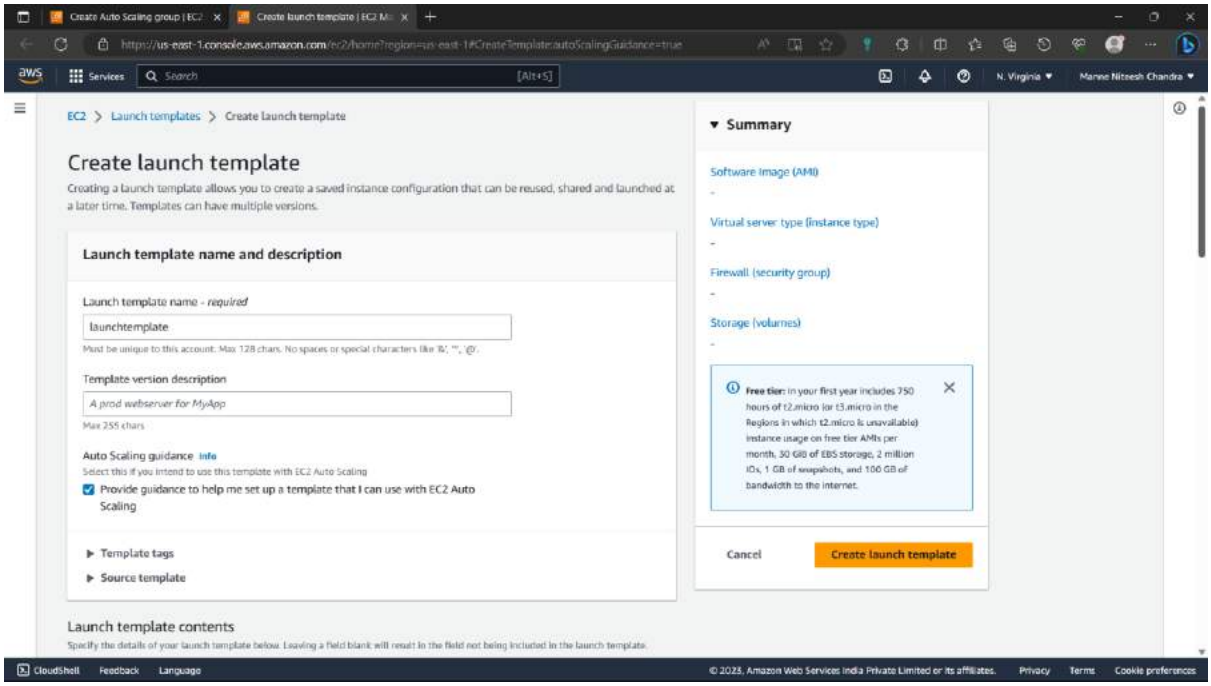
247)Click on “create auto scaling group” option.



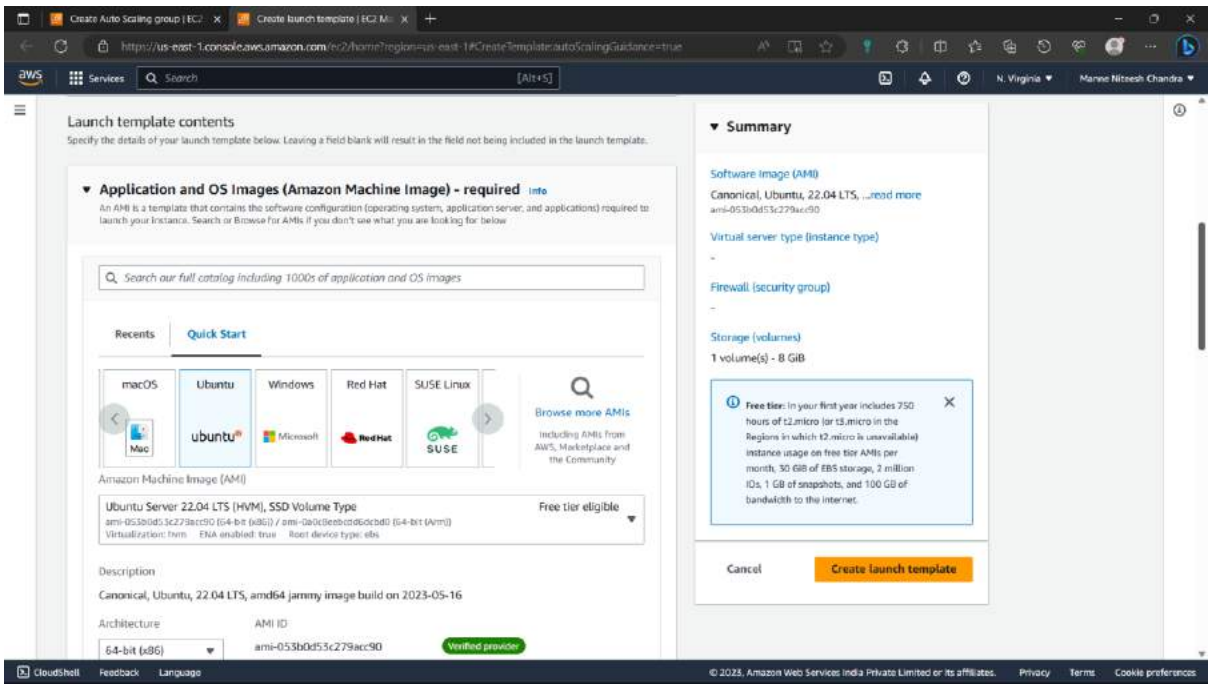
248)Give a name to the auto scaling and click on “create a launch template” in “Launch template” option.



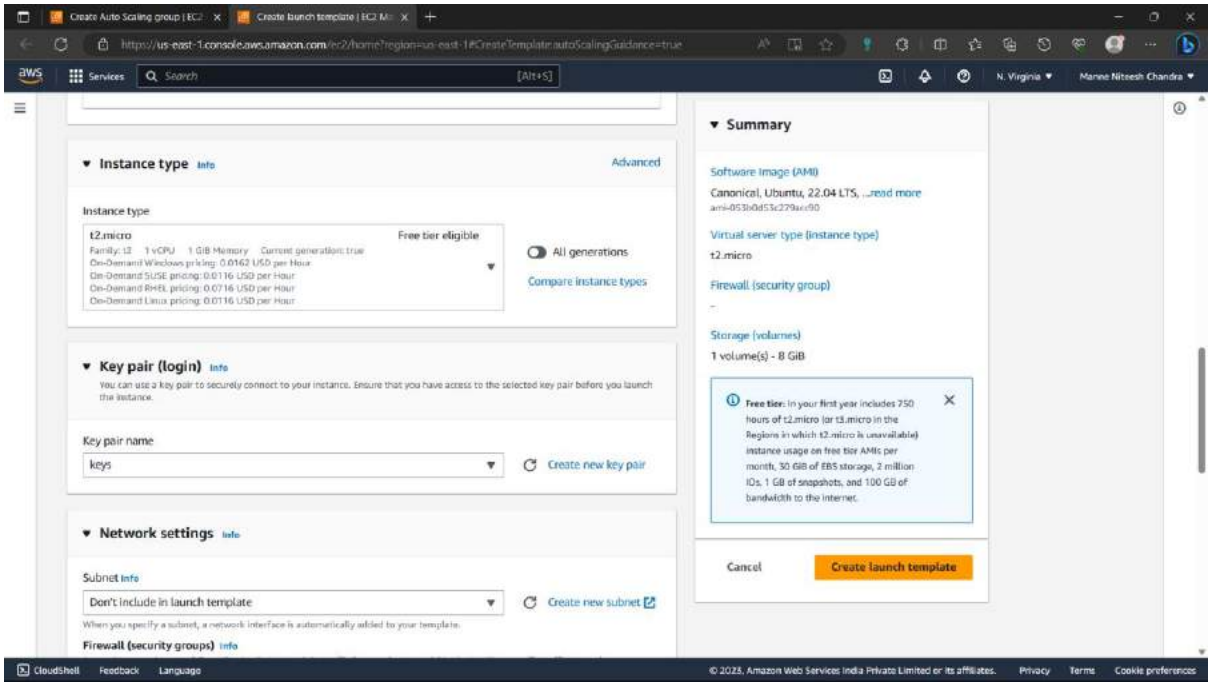
249)The launch template creation is opened in a new tab. Give a name to the launch template.



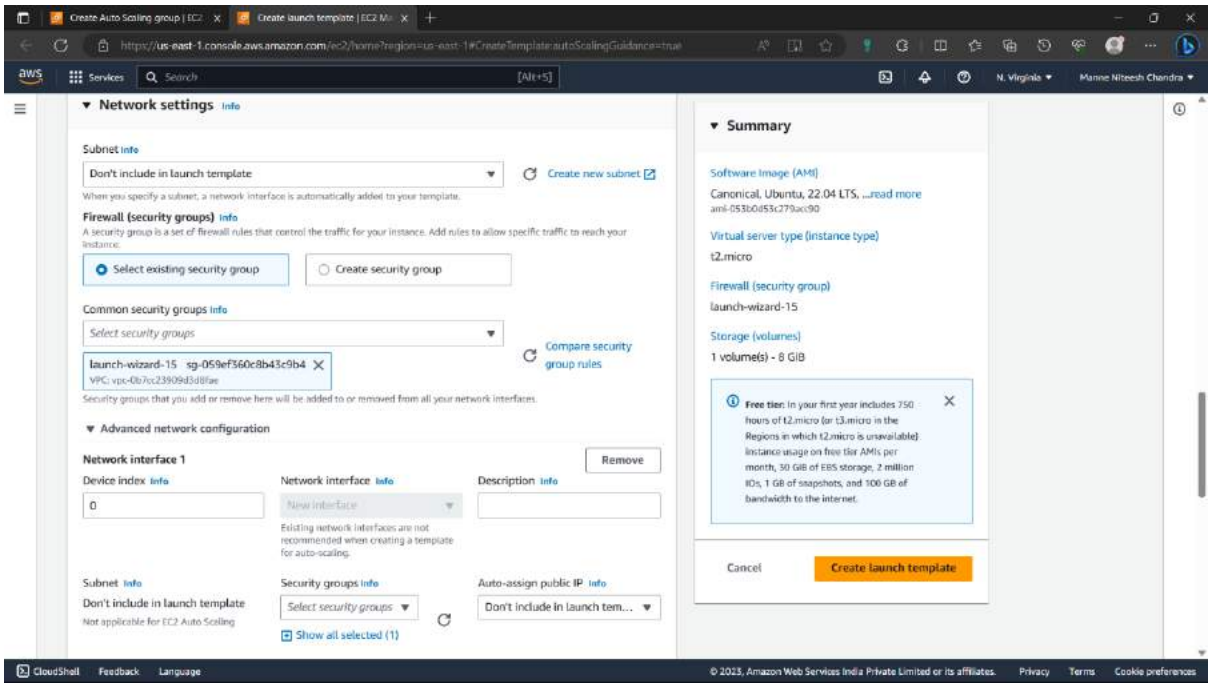
250)Select the AMI “Ubuntu” let everything be default.



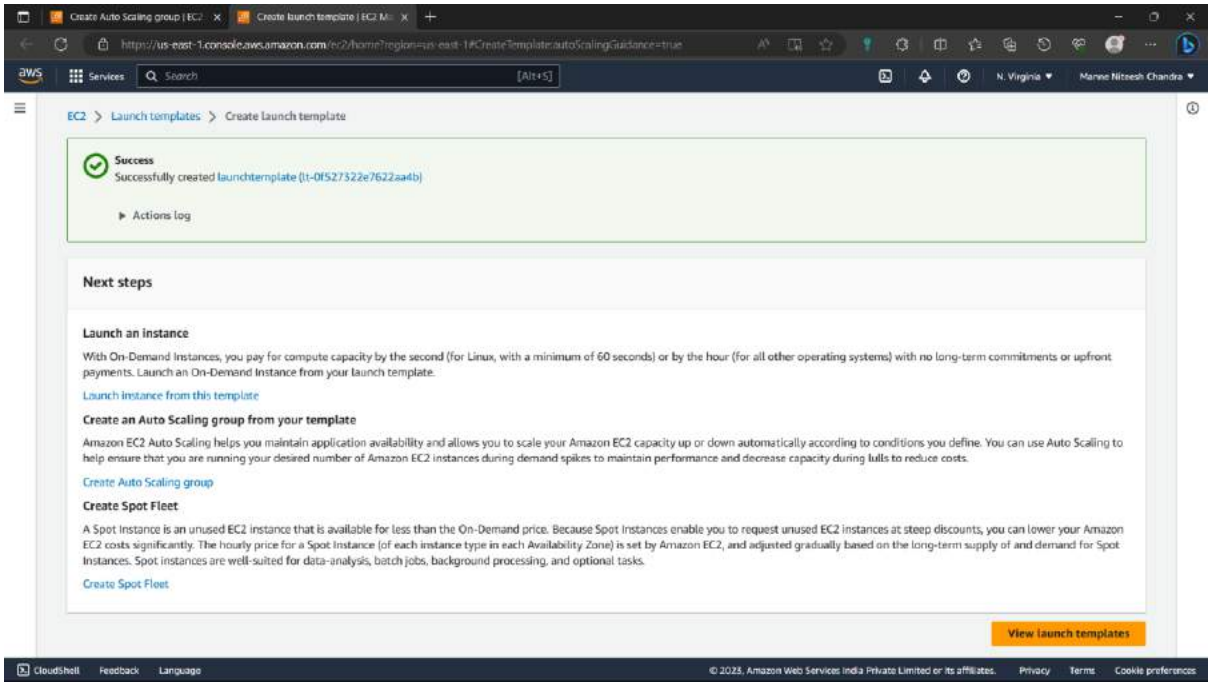
251)Instance type-“t2.micro”,key pair-keys(you have in your machine I.e., you created).



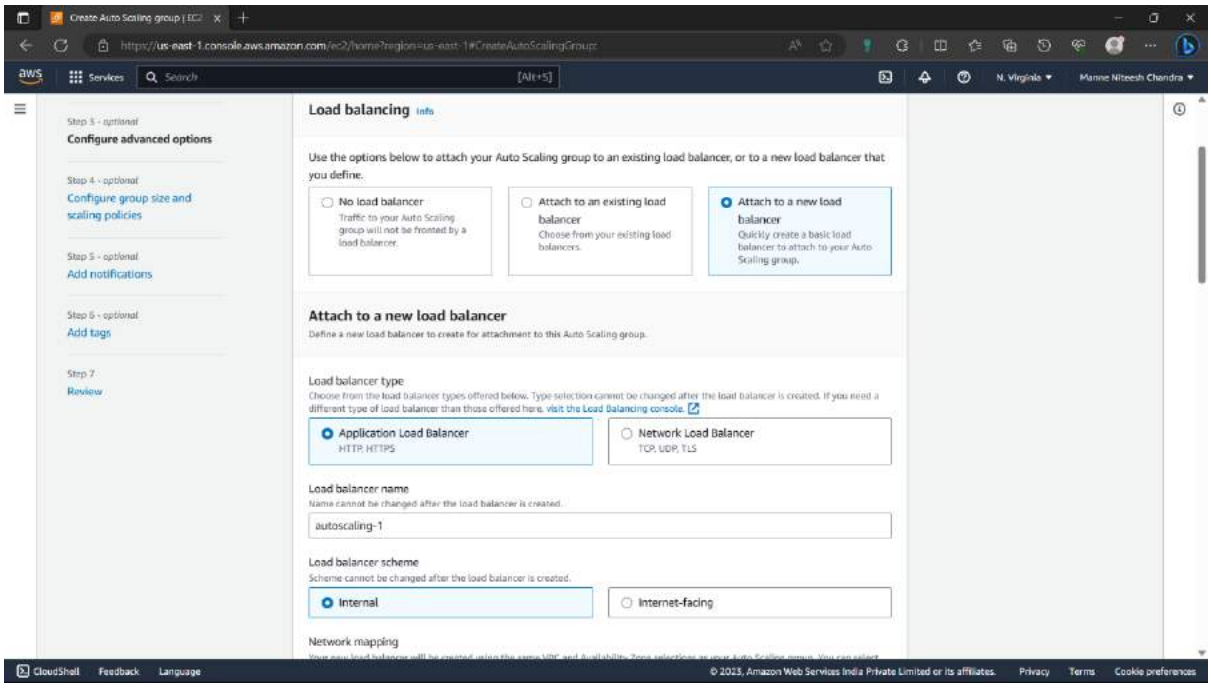
252)In network settings no ned to select the subnet let everything be default and click on “Create launch template”.



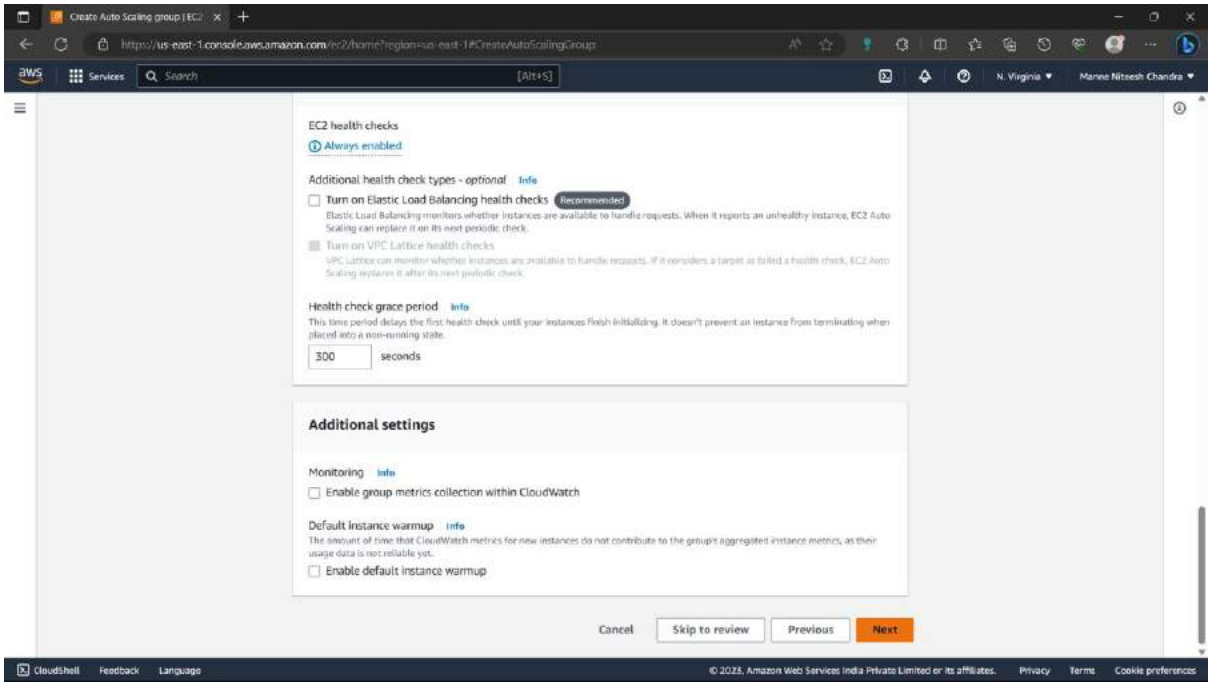
253)Launch template is created.



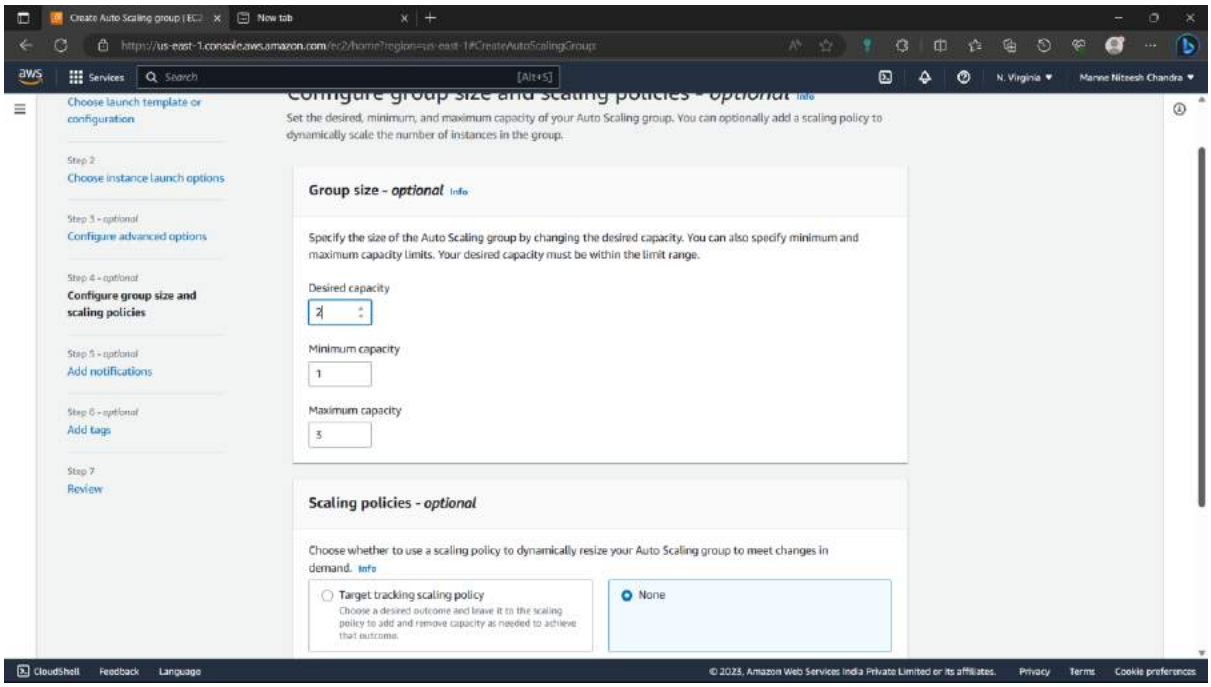
254)Go back to the auto scaling tab and select the created launch template and click on “next”. Select “Attach to a new load balancer” because I didn’t have a load balancer. If you have a load balancer attach it.



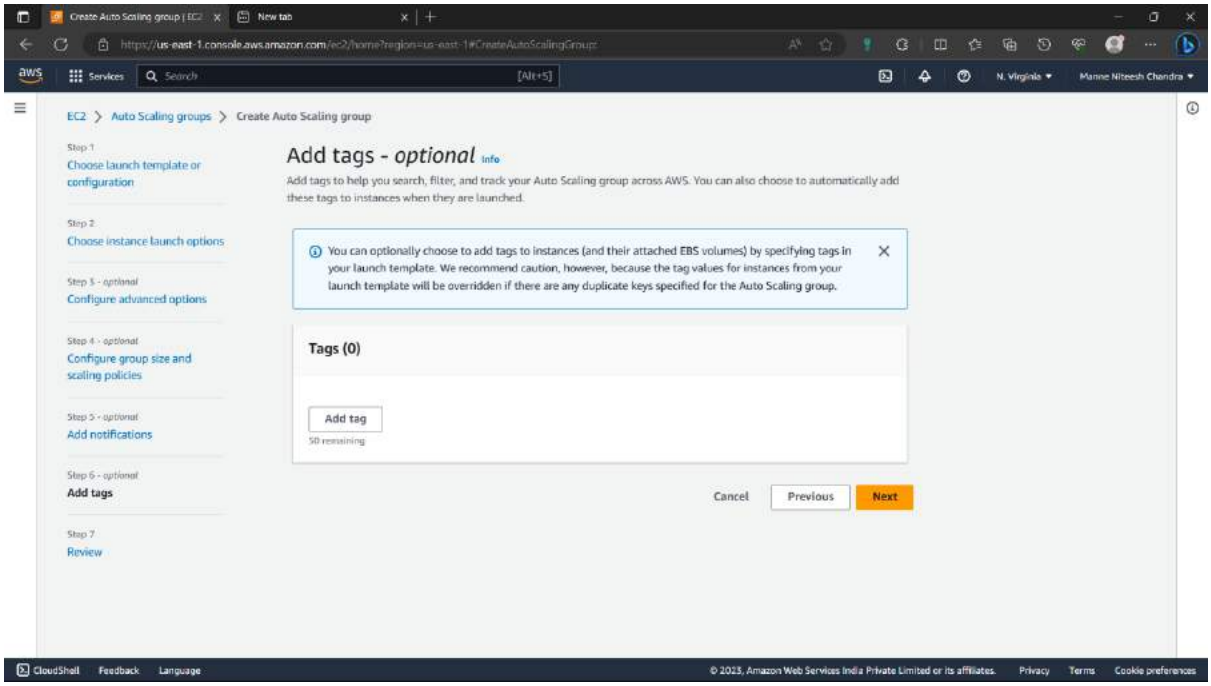
255)Click on “Next”.



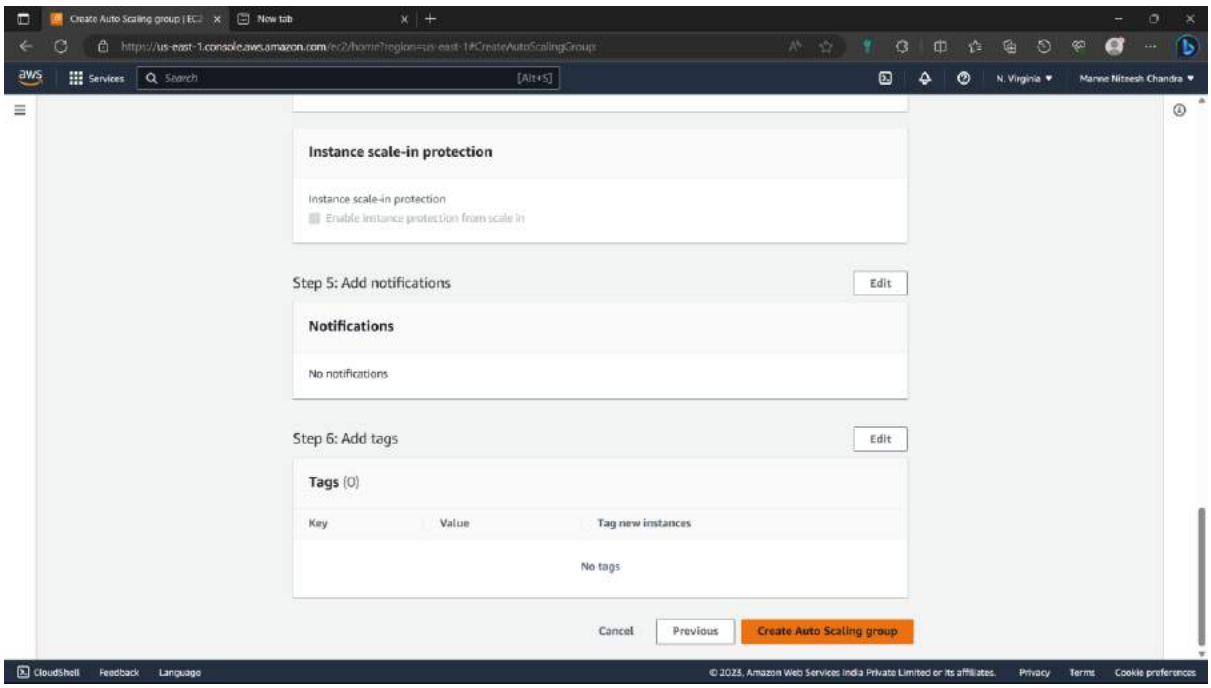
256)Give Desired, Maximum, Minimum capacity values (your choice) and click on “Next”.



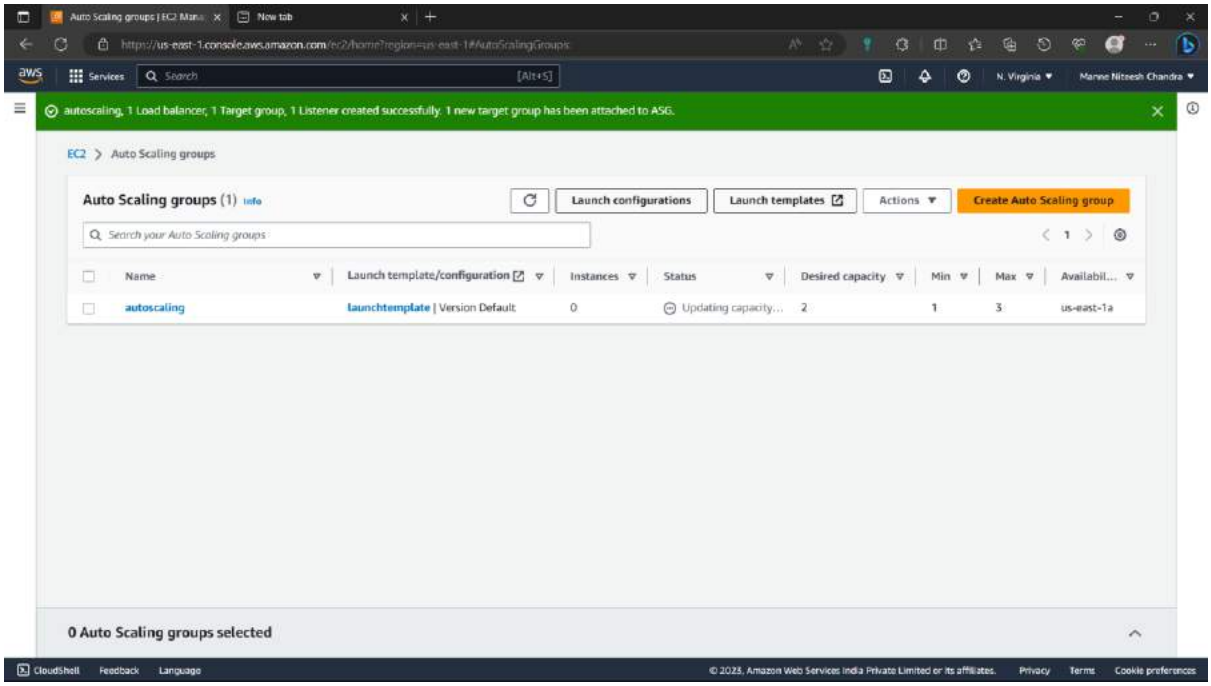
257)Click on “next” for step-5,6.



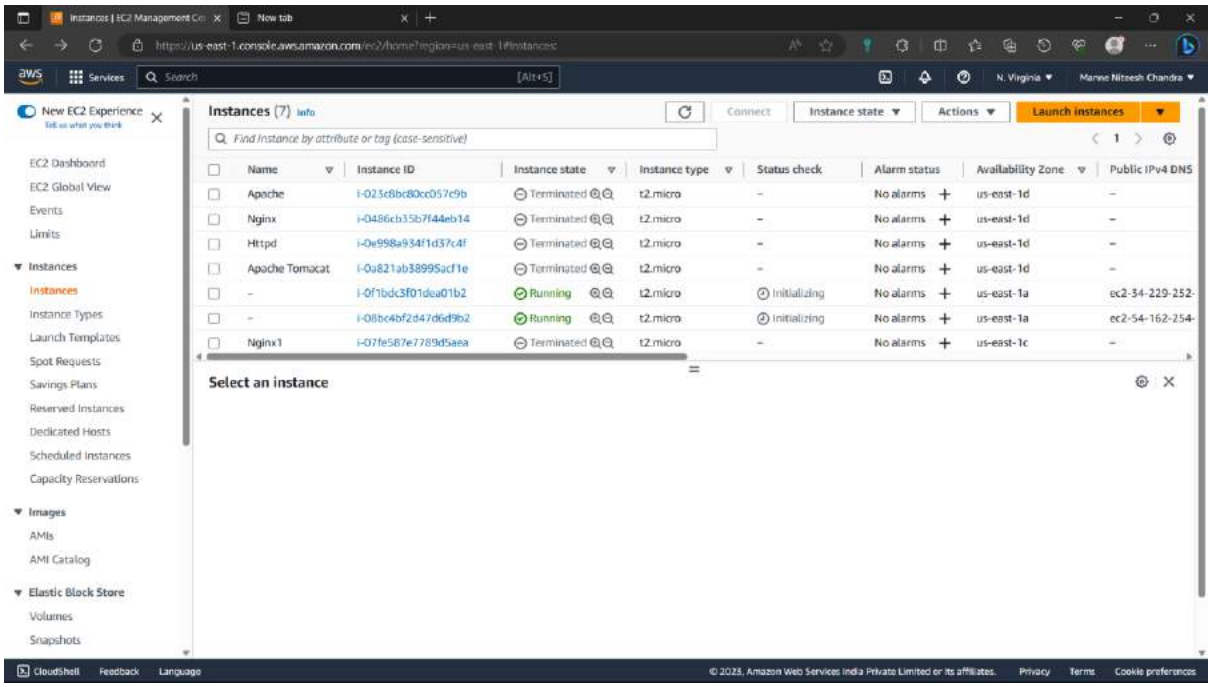
258)Review all the details and click on “Create Auto Scaling Group”.



259)The Auto Scaling Group is created.

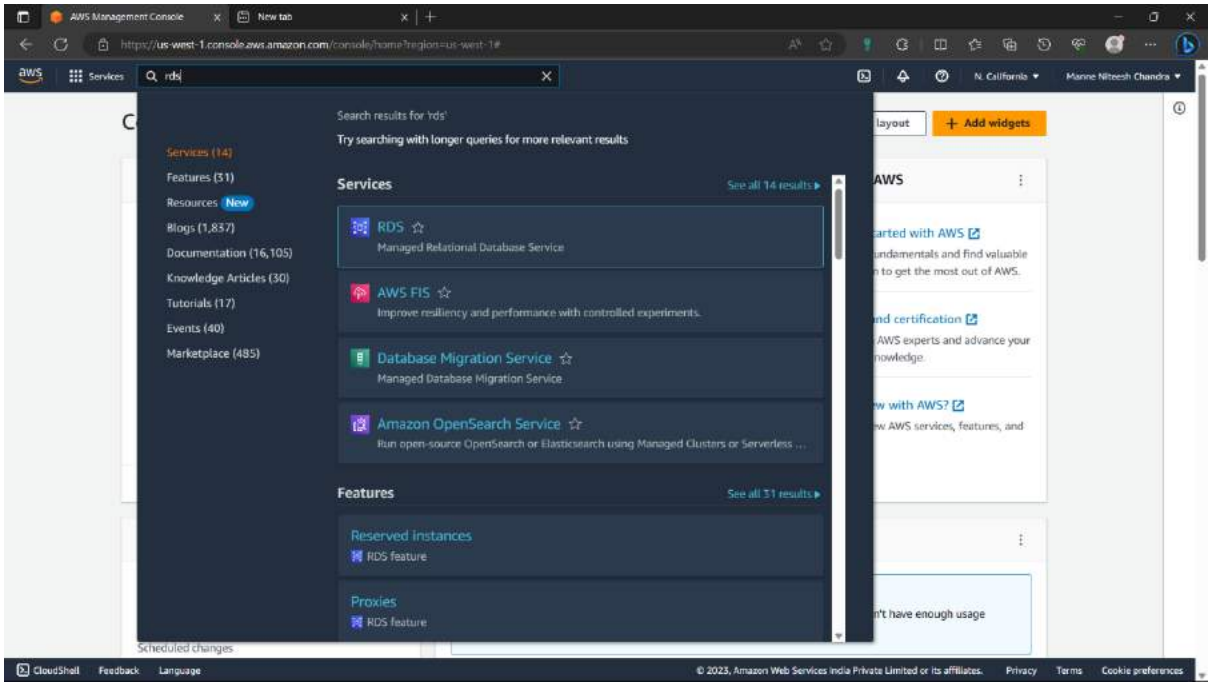


260)Go to “Instances” you can see the instances running(Instances without names) according to the capacity of users.

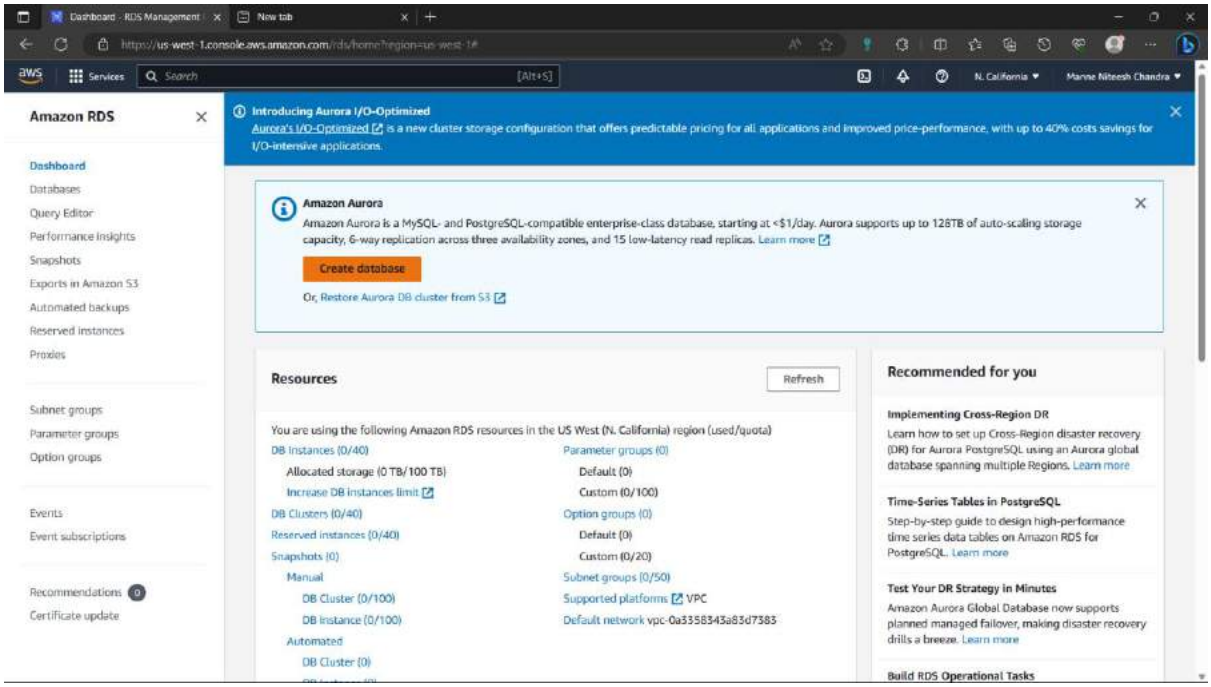


RDS:

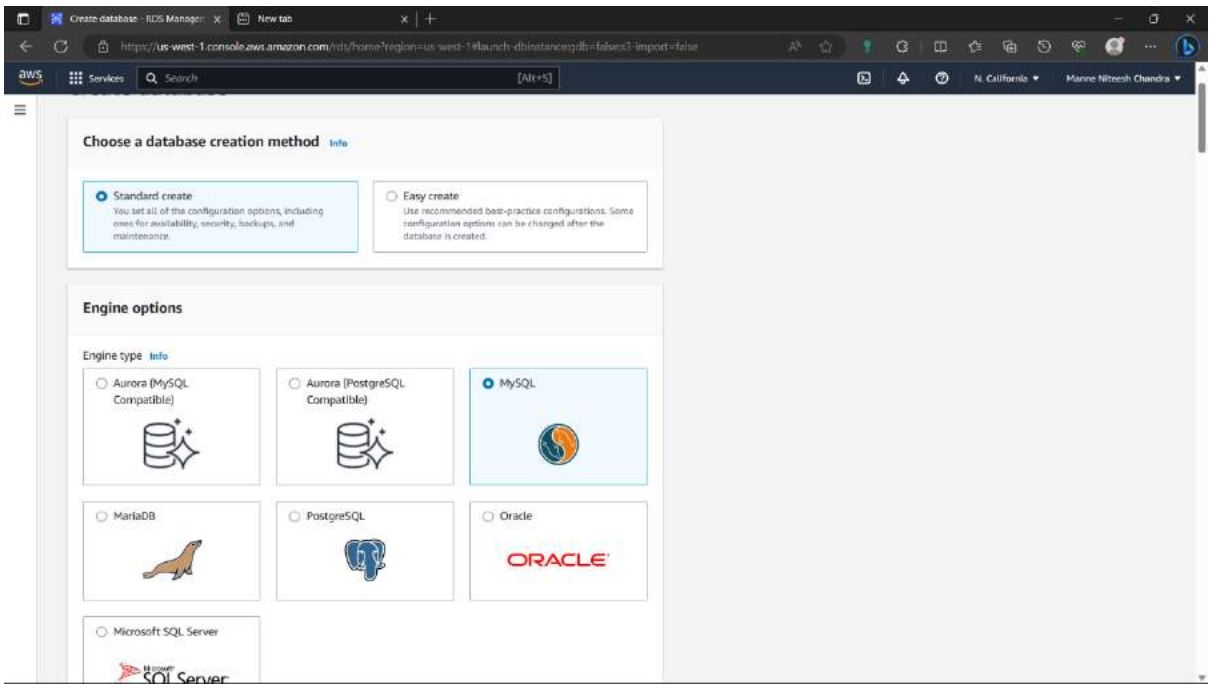
261)Type “RDS” in search bar and select the “RDS” option from below.



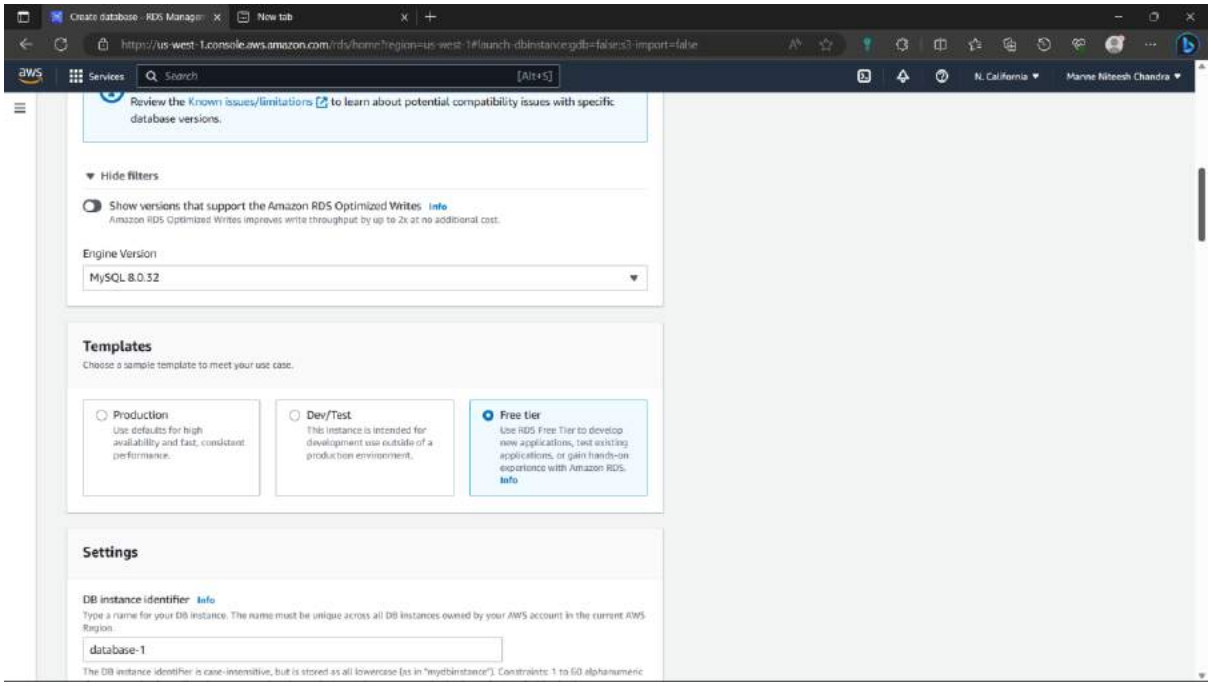
262)Click on “Create database” option.



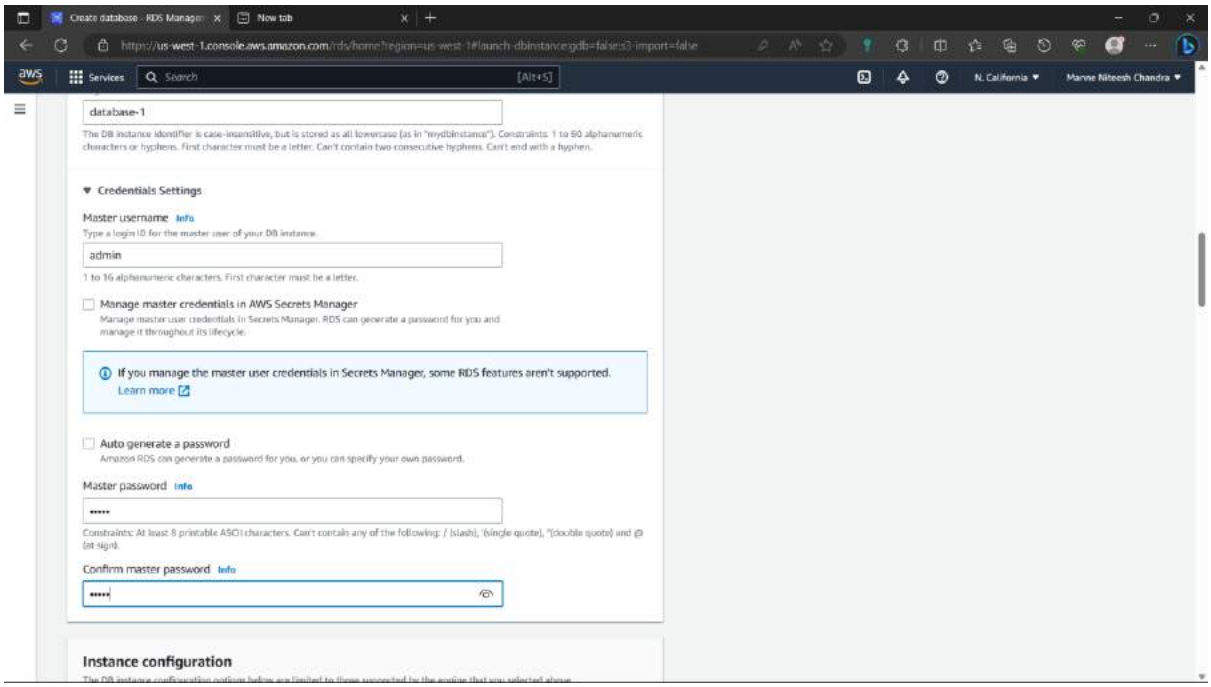
263)database creation method-“standard method”, Engine option- “MySql”.



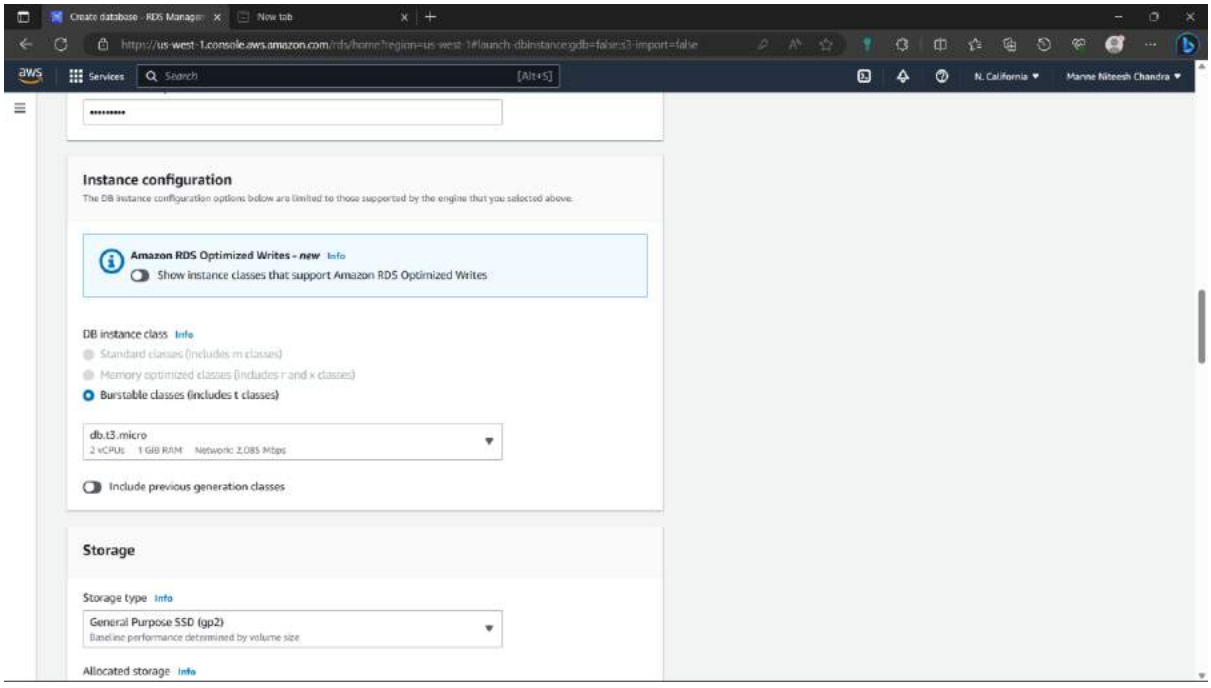
264)Template-“Free tier”.



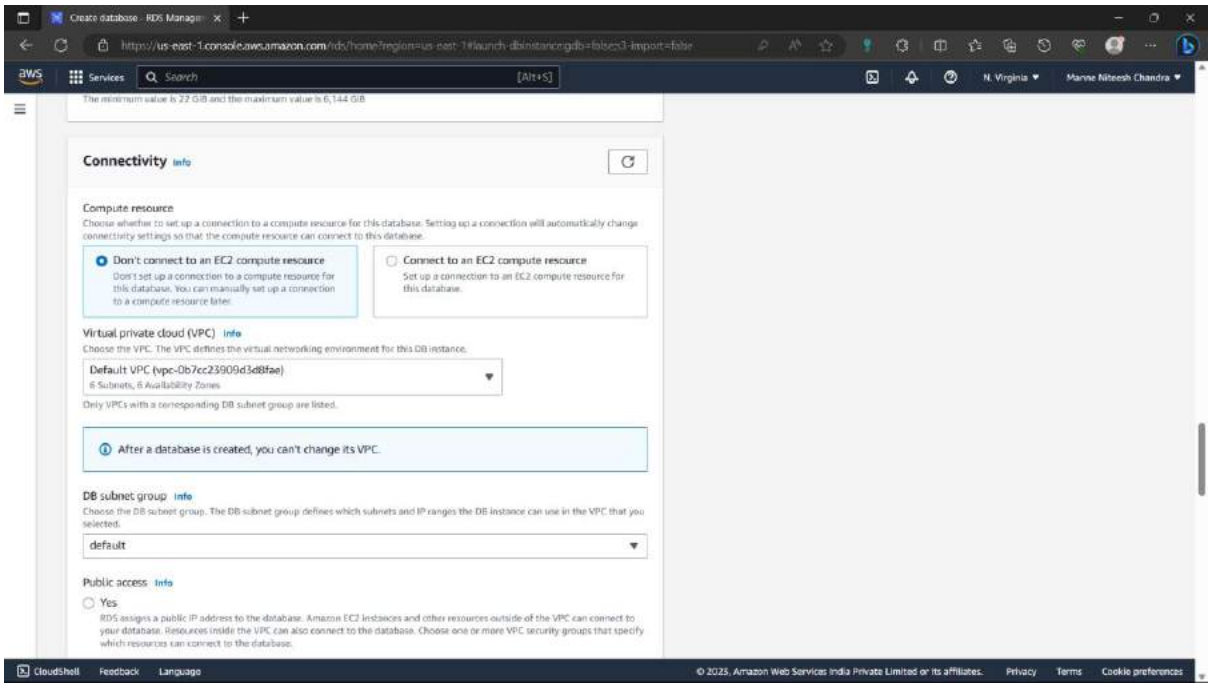
265)Give a name to the database and give a password to the admin in “credential settings”.



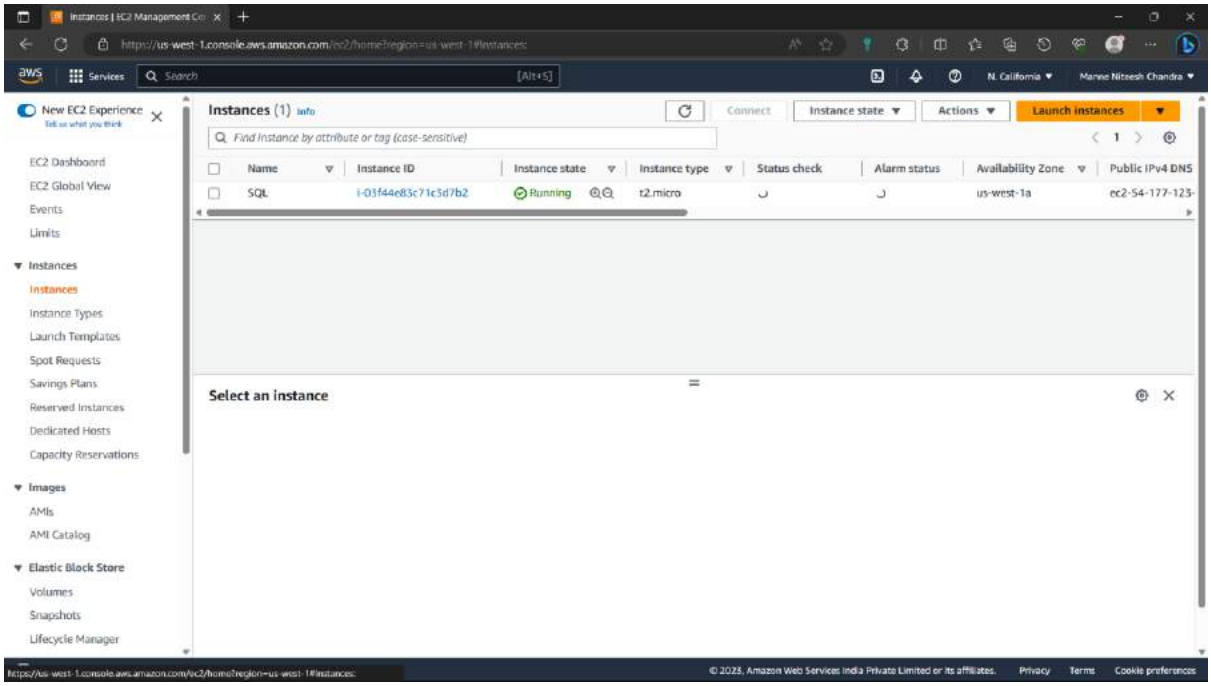
257)Instance configuration-“Burstable class”, Storage type- “General purpose”.



258)Connectivity-Don’t connect to EC2(you’re choice), Select VPC and let everything be default and click on “Create database”.

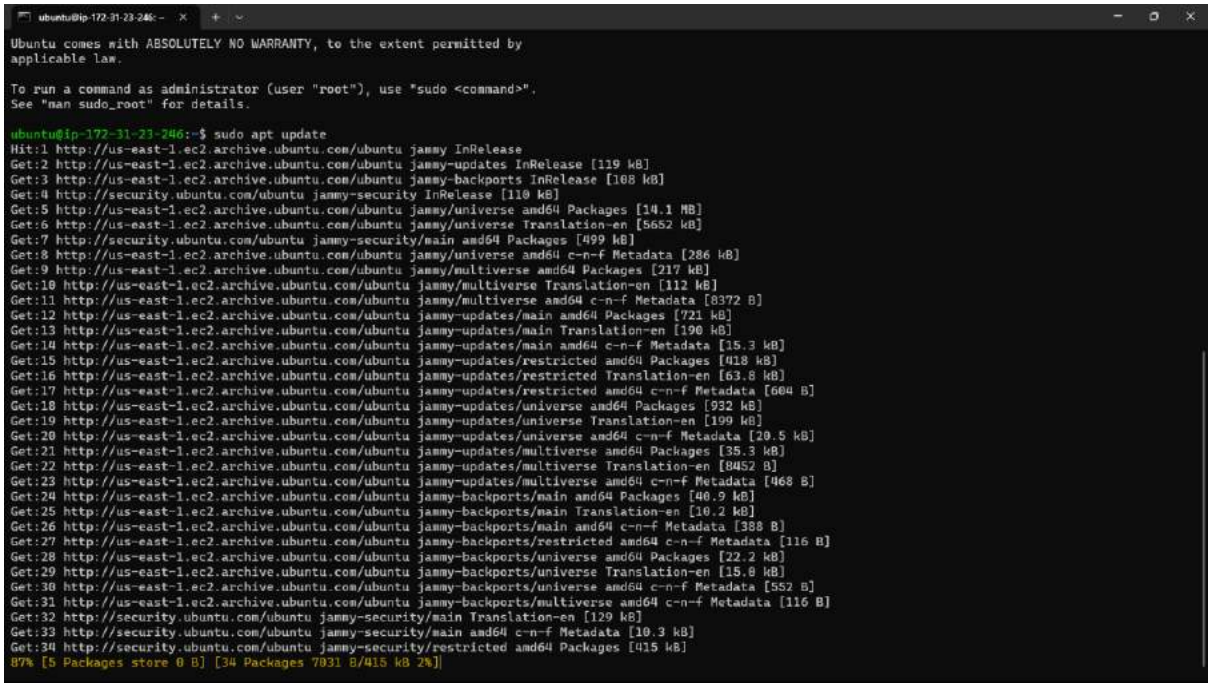


259)Now open the EC2 instance in the vpc (given in database) if you don’t have the VPC, Create the vpc by following the steps-(129-140).

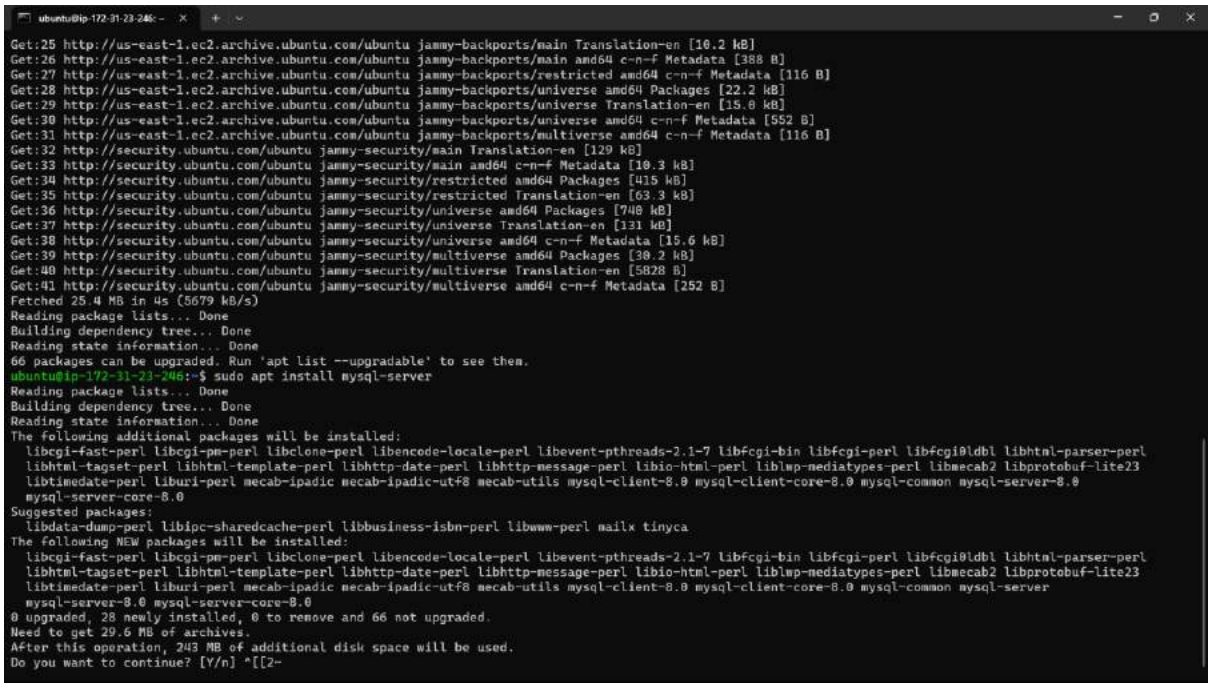


260)Access the instance by following the instructions from steps:(141-146). And type the command

“sudo apt update”: To install the terminal to latest version.



261)”sudo apt install mysql-server”: To install mysql in this instance.



262)”sudo sustemctl start mysql”: To start mysql

“sudo systemctl enable mysql”: To enable mysql

“sudo mysql”: To perform mysql programming in the instance.