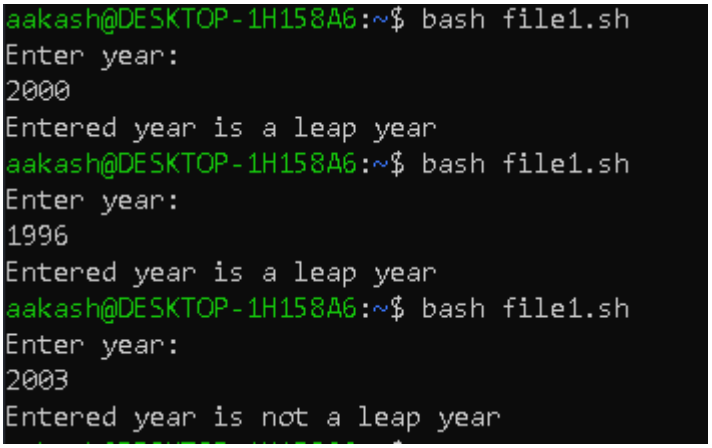
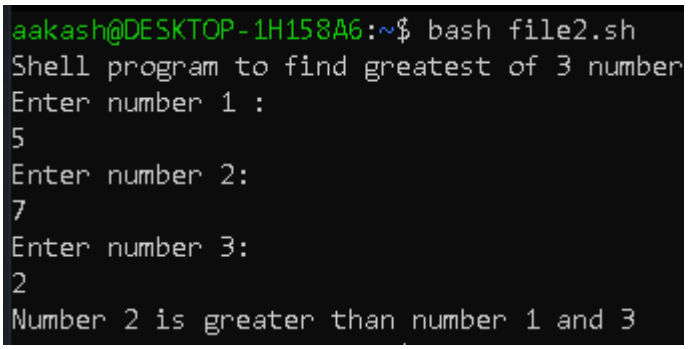
1.

```
#!/bin/bash

echo "Enter year: "
read y


if [[ $y%4 -eq 0 ]]
then

 if [[ $y%100 -eq 0 ]]
 then
    if [[ $y%400 -eq 0 ]]
    then
    echo "Entered year is a leap year"
    else
    echo "Entered year is not a leap year"
    fi
 else
 echo "Entered year is a leap year"
 fi

else
echo "Entered year is not a leap year"
fi
```



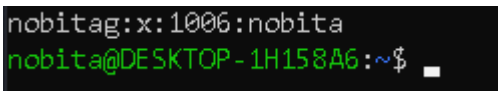
2.

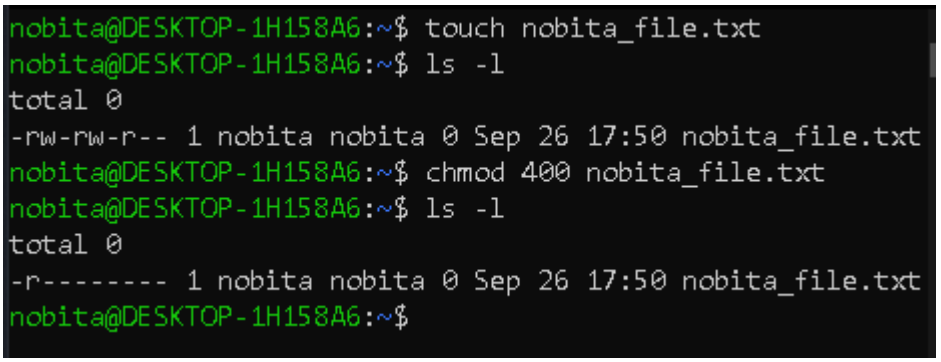
```
#!/bin/bash
echo "Shell program to find greatest of 3 number"

echo "Enter number 1 : "
read a

echo "Enter number 2: "
read b

echo "Enter number 3: "
read c


if [[ $a -gt $b ]]
then
if [[ $a -gt $c ]]
then
echo "Number 1 is greater than number 2 and 3"
else
echo "Number 3 is greater than number 1 and 2"
fi
else
 if [[ $b -gt $c ]]
 then
 echo "Number 2 is greater than number 1 and 3"
 else
 echo "Number 3 is greater than number 1 and 2"
 fi
fi
```



3.

#adduser nobita
#groupadd nobitag
#usermod -a -G nobitag nobita



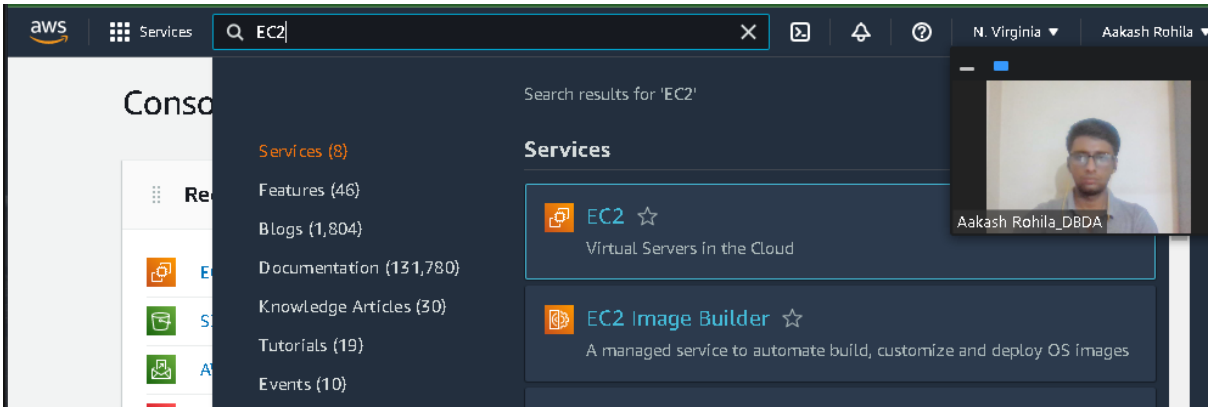


4.

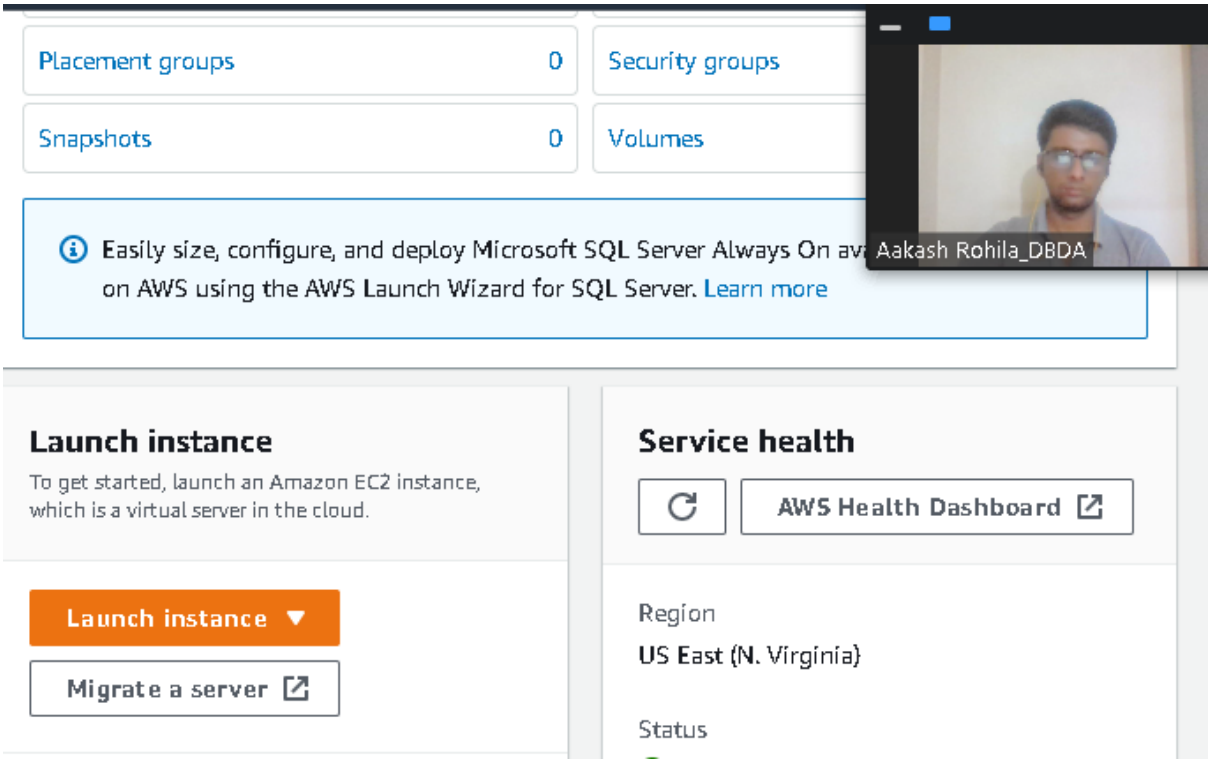
```
aakash@DESKTOP-1H158A6:~$ mkdir dbda
aakash@DESKTOP-1H158A6:~$ cd dbda
aakash@DESKTOP-1H158A6:~/dbda$ touch Sep.txt
aakash@DESKTOP-1H158A6:~/dbda$ ls
Sep.txt
aakash@DESKTOP-1H158A6:~/dbda$ mv Sep.txt Batch2.txt
aakash@DESKTOP-1H158A6:~/dbda$ ls
Batch2.txt
aakash@DESKTOP-1H158A6:~/dbda$ cd ..
aakash@DESKTOP-1H158A6:~$ mkdir dbda_B2
aakash@DESKTOP-1H158A6:~$ tree
.
├── dbda
│   └── Batch2.txt
├── dbda_B2
├── file1.sh
└── file2.sh

2 directories, 3 files
aakash@DESKTOP-1H158A6:~$ mv dbda/Batch2.txt dbda_B2
aakash@DESKTOP-1H158A6:~$ tree
.
├── dbda
├── dbda_B2
│   └── Batch2.txt
├── file1.sh
└── file2.sh

2 directories, 3 files
```

5.

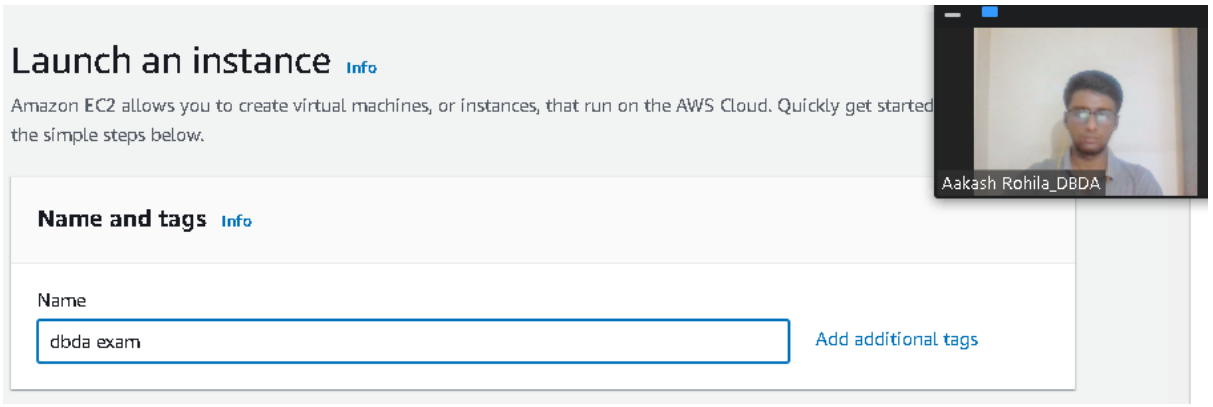
Step1>>
Go to EC2 service on Amazon AWS



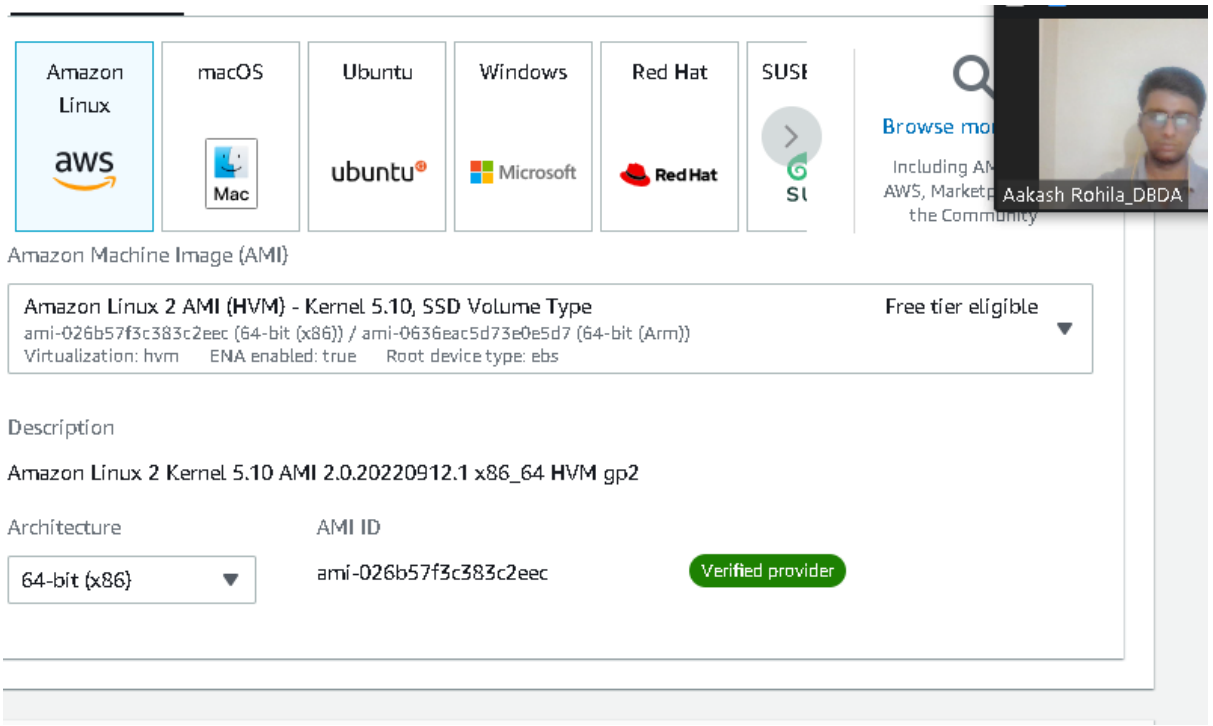
Step2>>
Launch an E2 instance



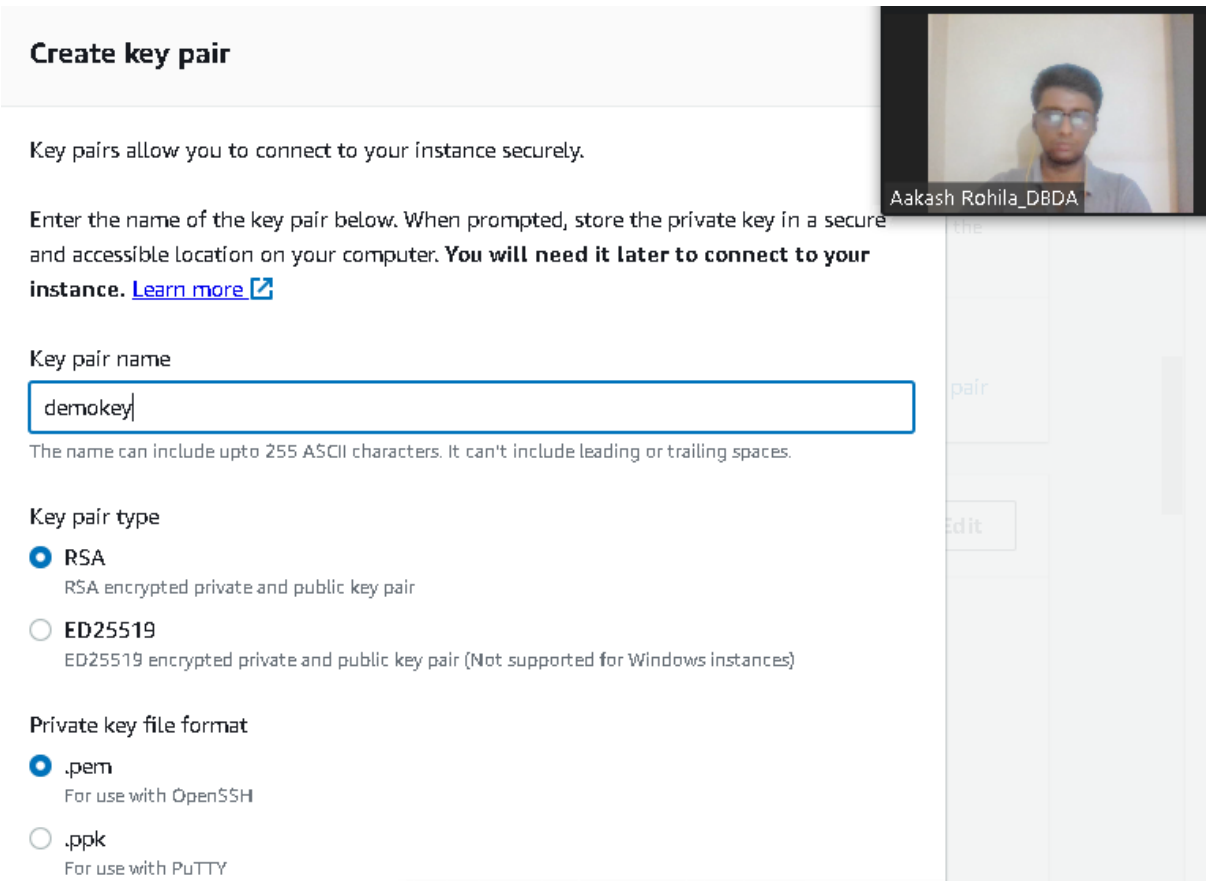
Step3>>Name your instance



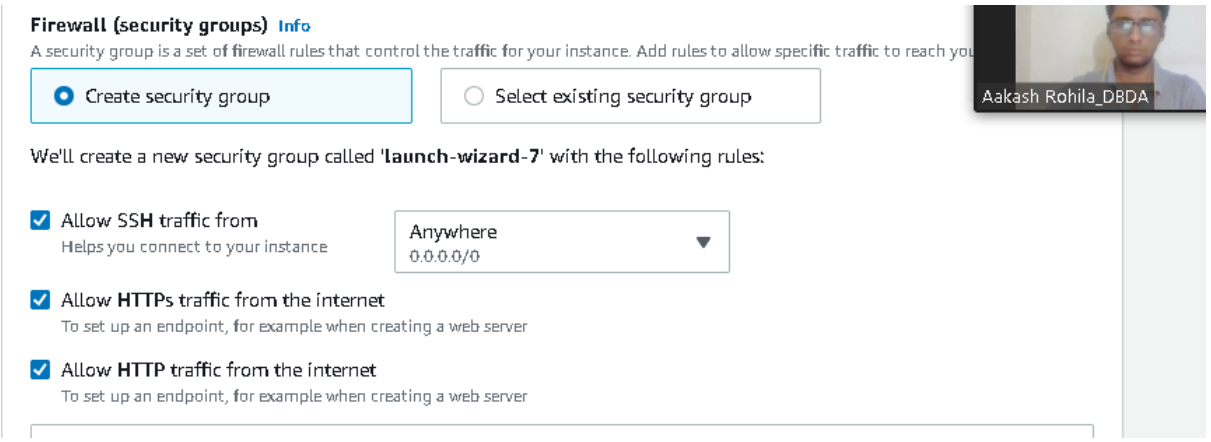
Step4>> Make sure you have selected the correct OS and version



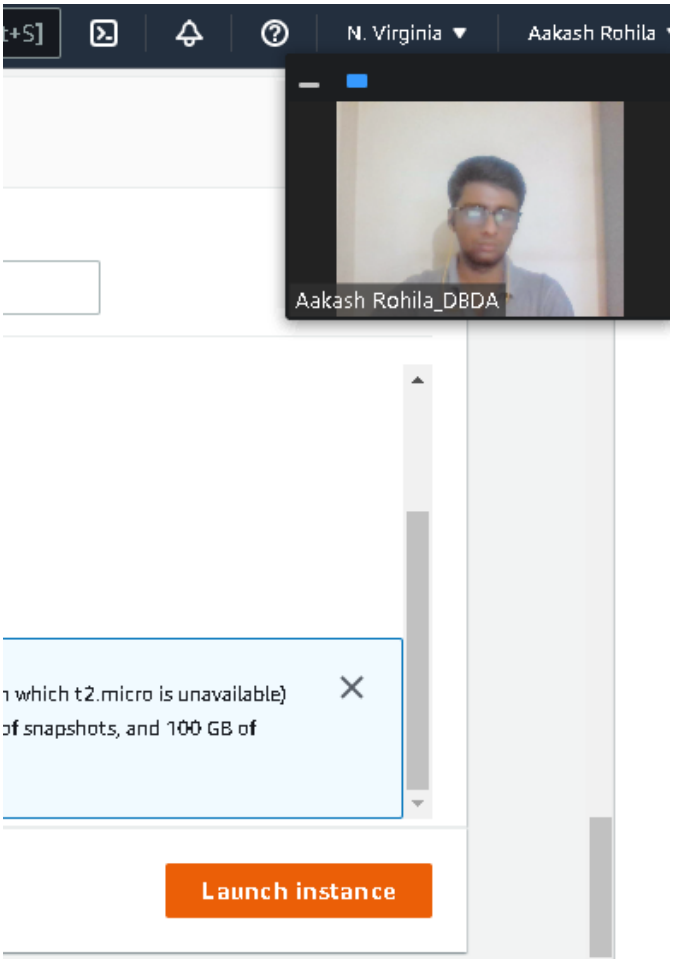
Step 5>>
Create new keypair with .pem extension for linux machines



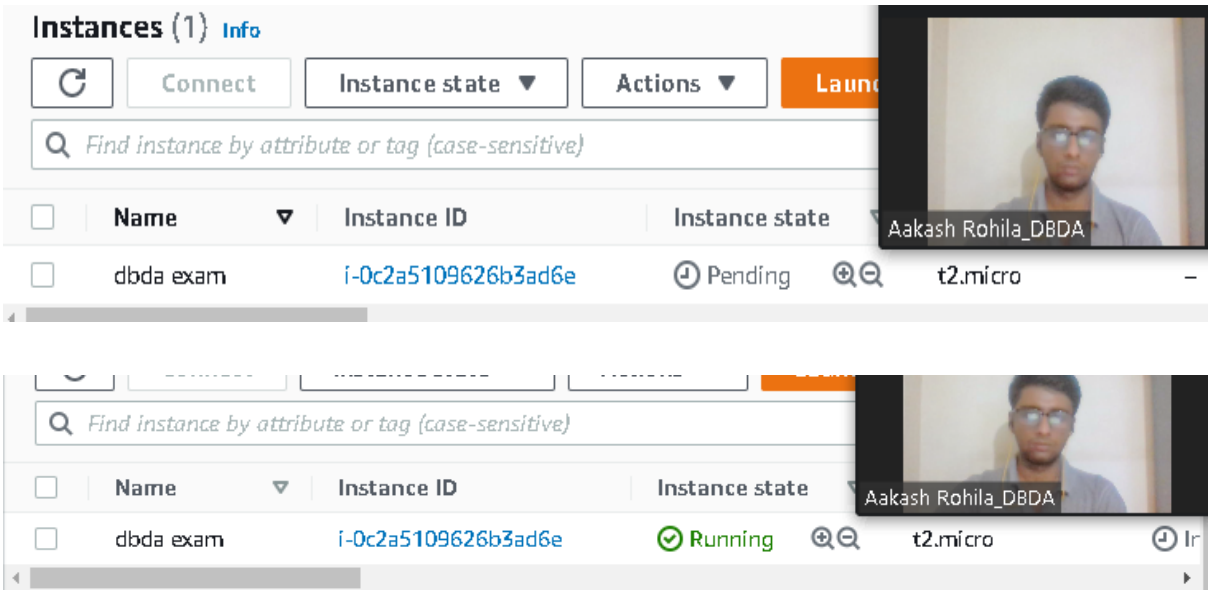
Step6>>Select traffic from all source



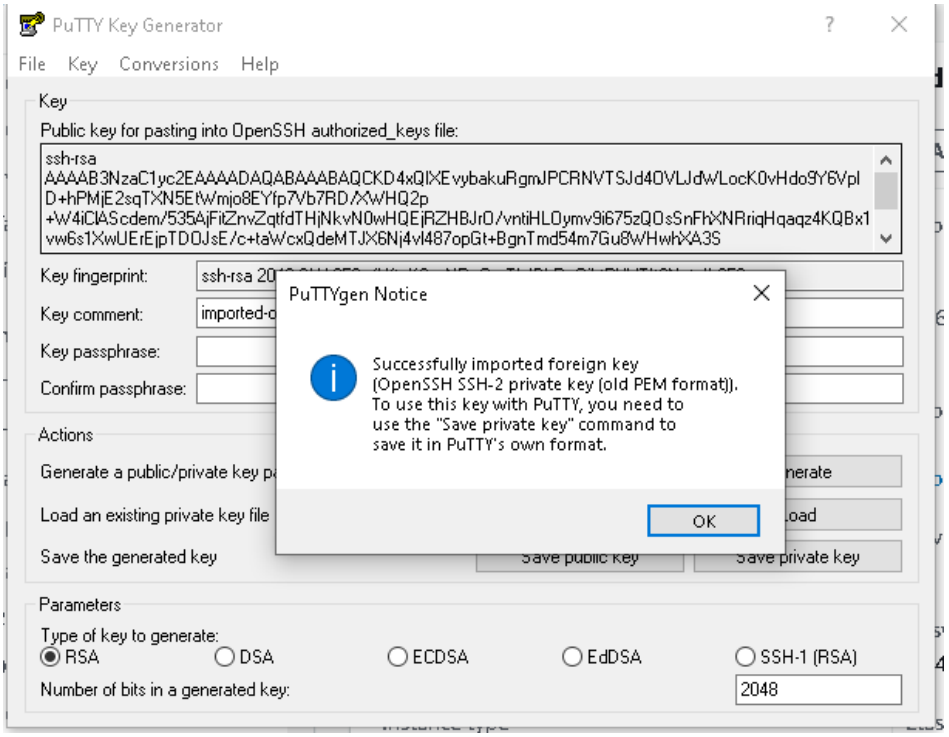
Step7>>Launch Instance



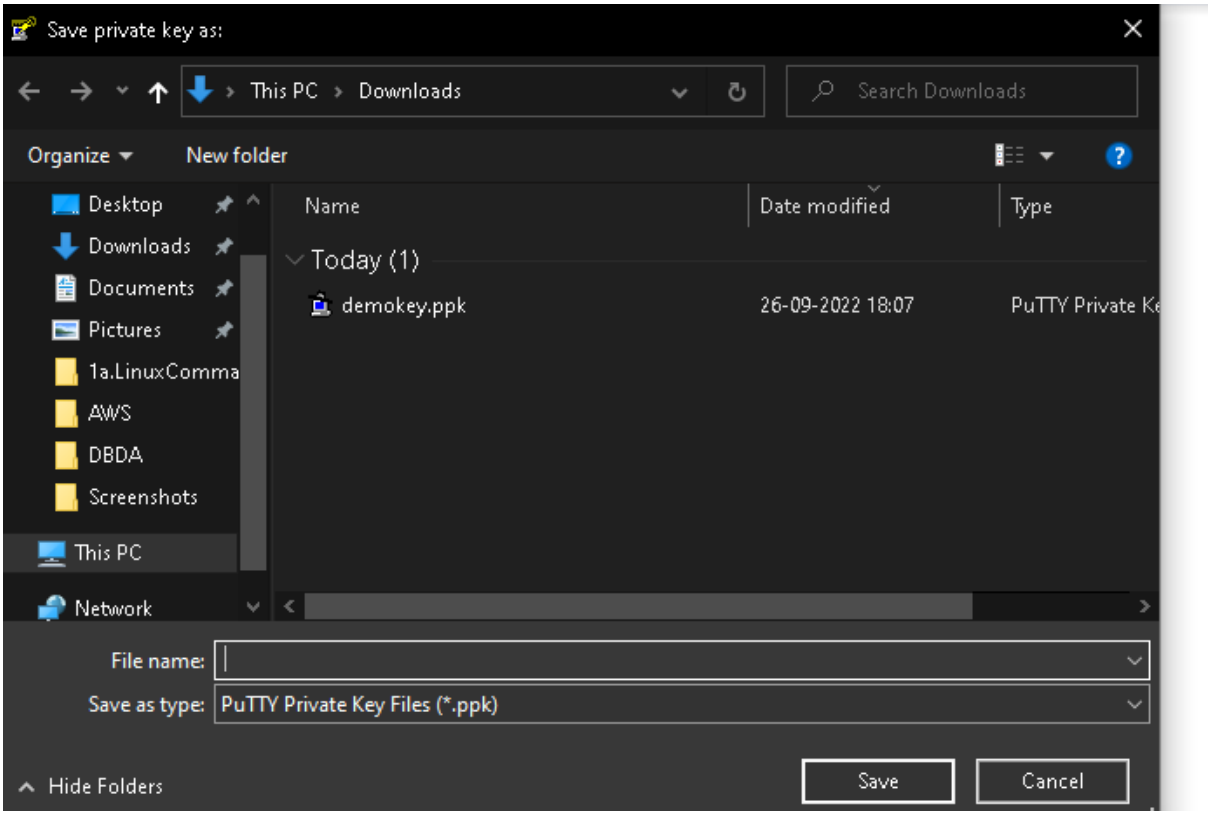
Step8>>Instance Created



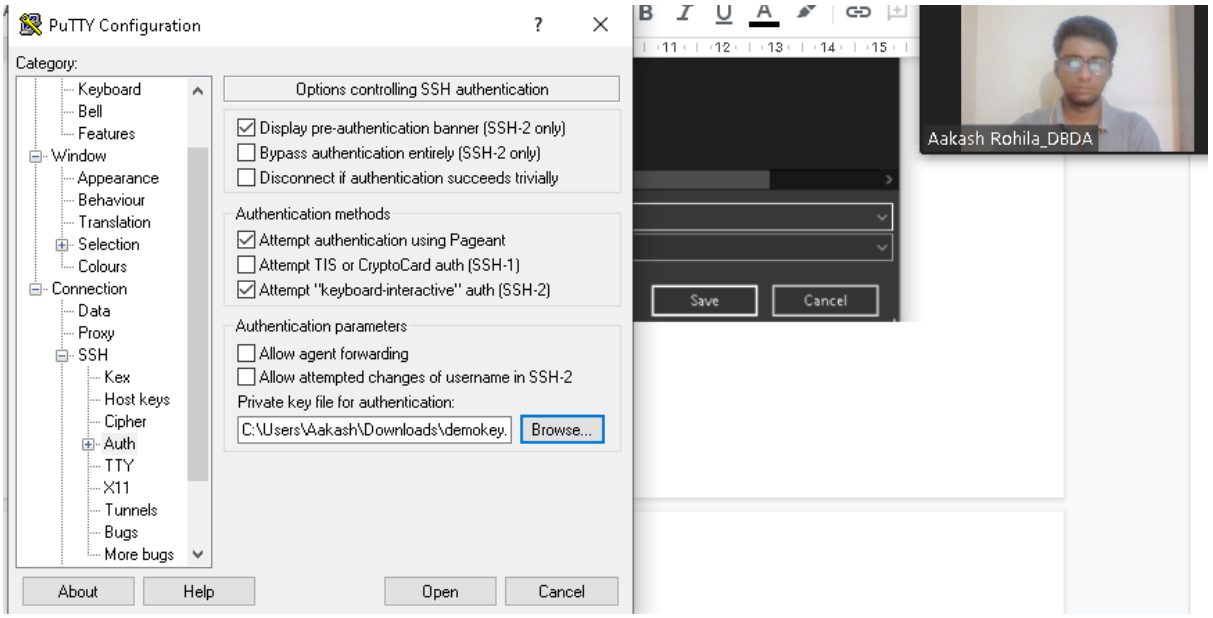
Step9>>Load .pem file in PuttyGen to convert to .ppk format



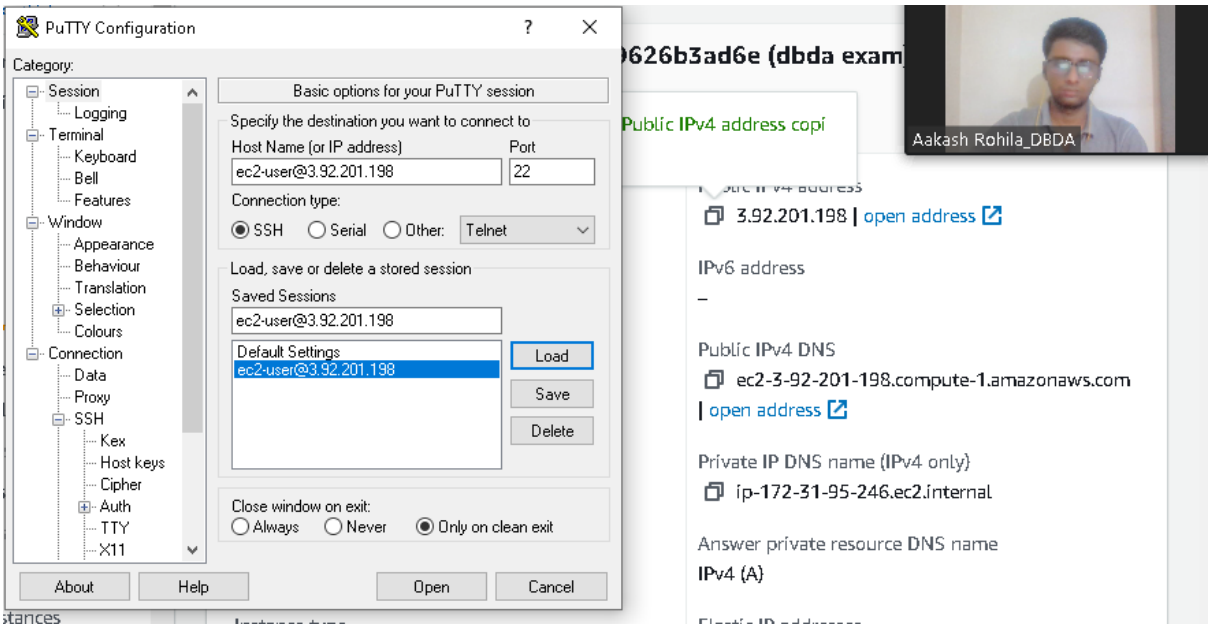
Step10>>Then Click on save private key and enter name of .ppk file and it should save your file in ..ppk format



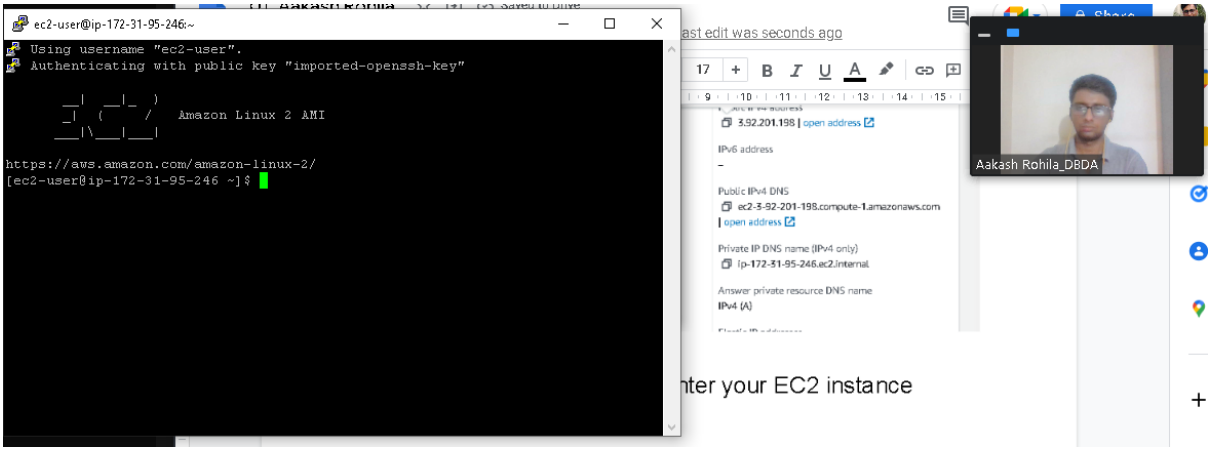
Step11>>Now start Putty on windows and load demokey.ppk file



Step12>>Now copy ip address from ec2 instance created and save it in your Putty config.



Step13>>Accept the next window and click enter your EC2 instance should start:



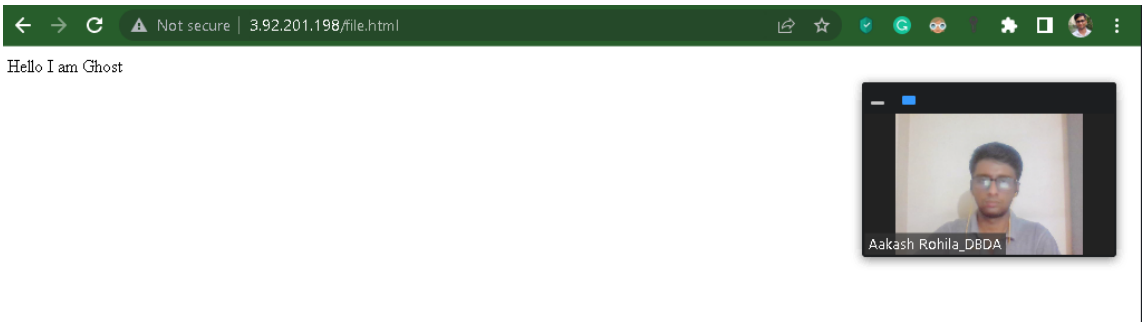

```
[root@ip-172-31-95-246 ec2-user]# yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                               | 3.7 kB     00:00
No packages marked for update
[root@ip-172-31-95-246 ec2-user]# yum install httpd -y
```

```
[root@ip-172-31-95-246 ec2-user]# systemctl start httpd
[root@ip-172-31-95-246 ec2-user]# systemctl enable httpd
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service
o /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-95-246 ec2-user]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor pres
: disabled)
   Active: active (running) since Mon 2022-09-26 12:47:45 UTC; 18s ago
     Docs: man:httpd.service(8)
 Main PID: 3468 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes
rved/sec:    0 B/sec"
   CGroup: /system.slice/httpd.service
           ├─3468 /usr/sbin/httpd -DFOREGROUND
           ├─3469 /usr/sbin/httpd -DFOREGROUND
           ├─3470 /usr/sbin/httpd -DFOREGROUND
           ├─3471 /usr/sbin/httpd -DFOREGROUND
           ├─3472 /usr/sbin/httpd -DFOREGROUND
           └─3473 /usr/sbin/httpd -DFOREGROUND

Sep 26 12:47:44 ip-172-31-95-246.ec2.internal systemd[1]: Starting The Apache.
Sep 26 12:47:45 ip-172-31-95-246.ec2.internal systemd[1]: Started The Apache .
Hint: Some lines were ellipsized, use -l to show in full.
```

```
[root@ip-172-31-95-246 ec2-user]# cd /var/www/html
[root@ip-172-31-95-246 html]# nano file.html
[root@ip-172-31-95-246 html]#
```

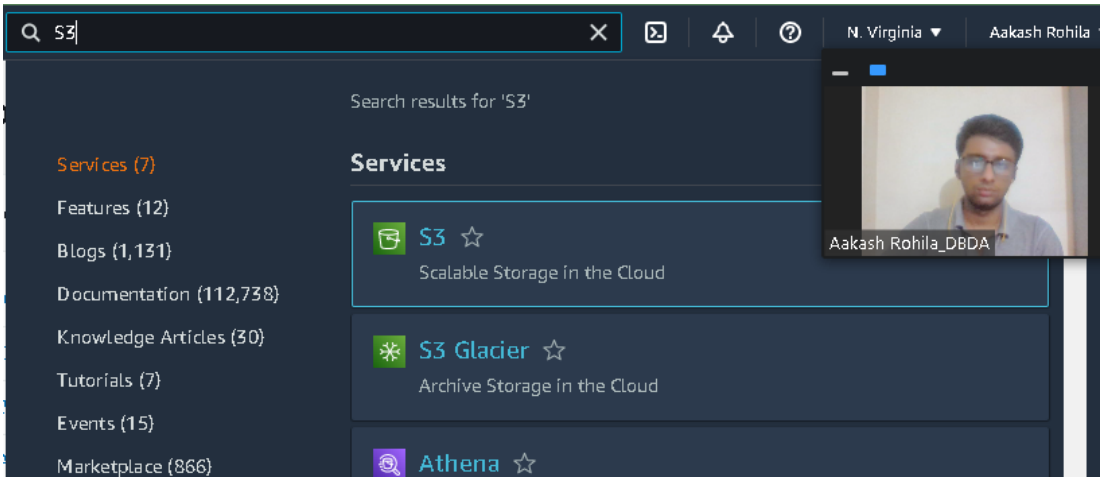
In the above command, we have started HTTP server and hosted our file.html context on it.

Step14>> Now enter IP address in the browser to check if our site is up and running

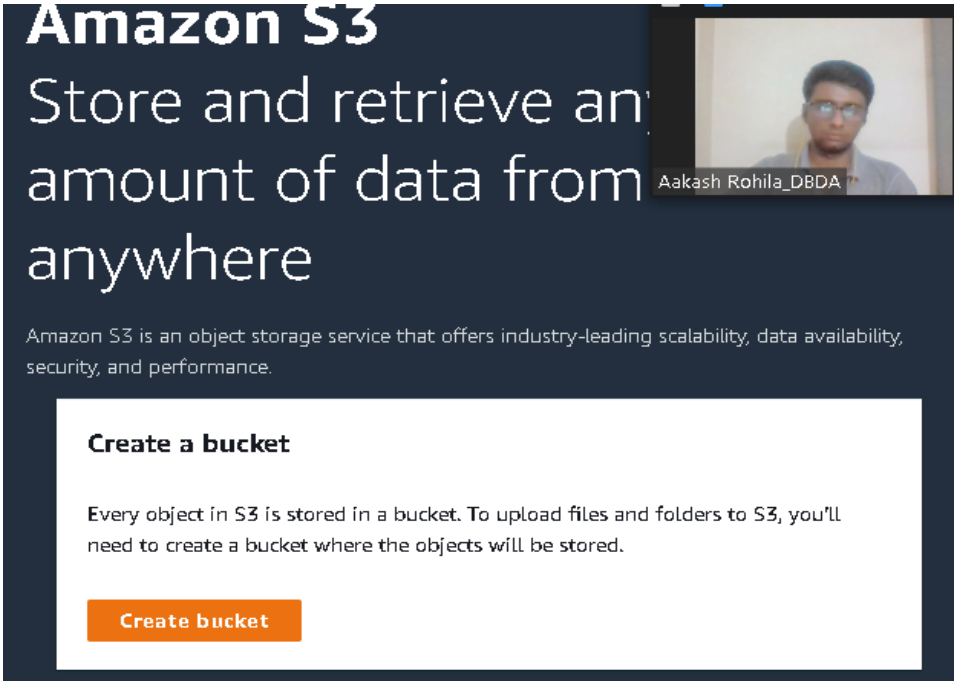


6.

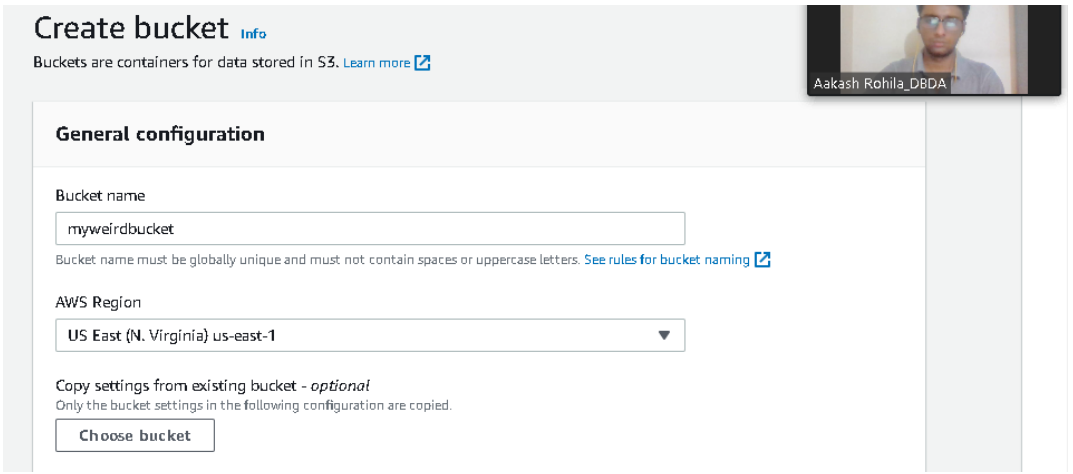
Step1>>Now to create an S3 bucket, go to AWS website and search for S3 and click on it…



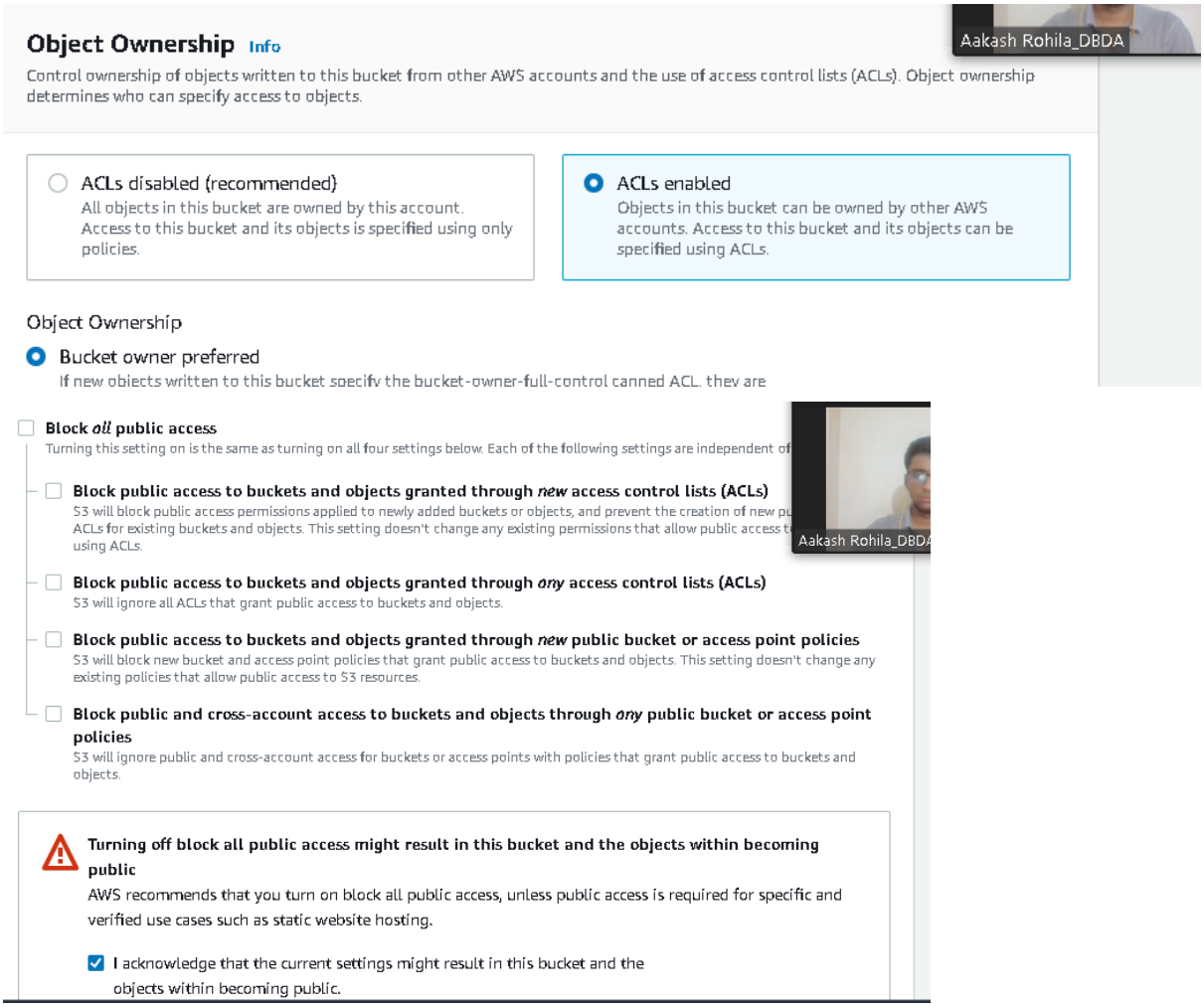
Step2>>Click on create bucket



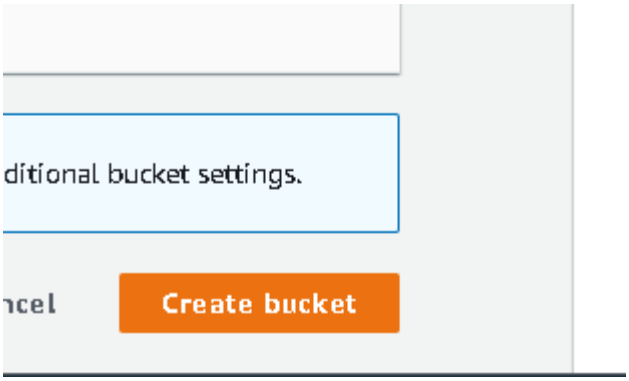
Step3>>Enter your unique bucket name…



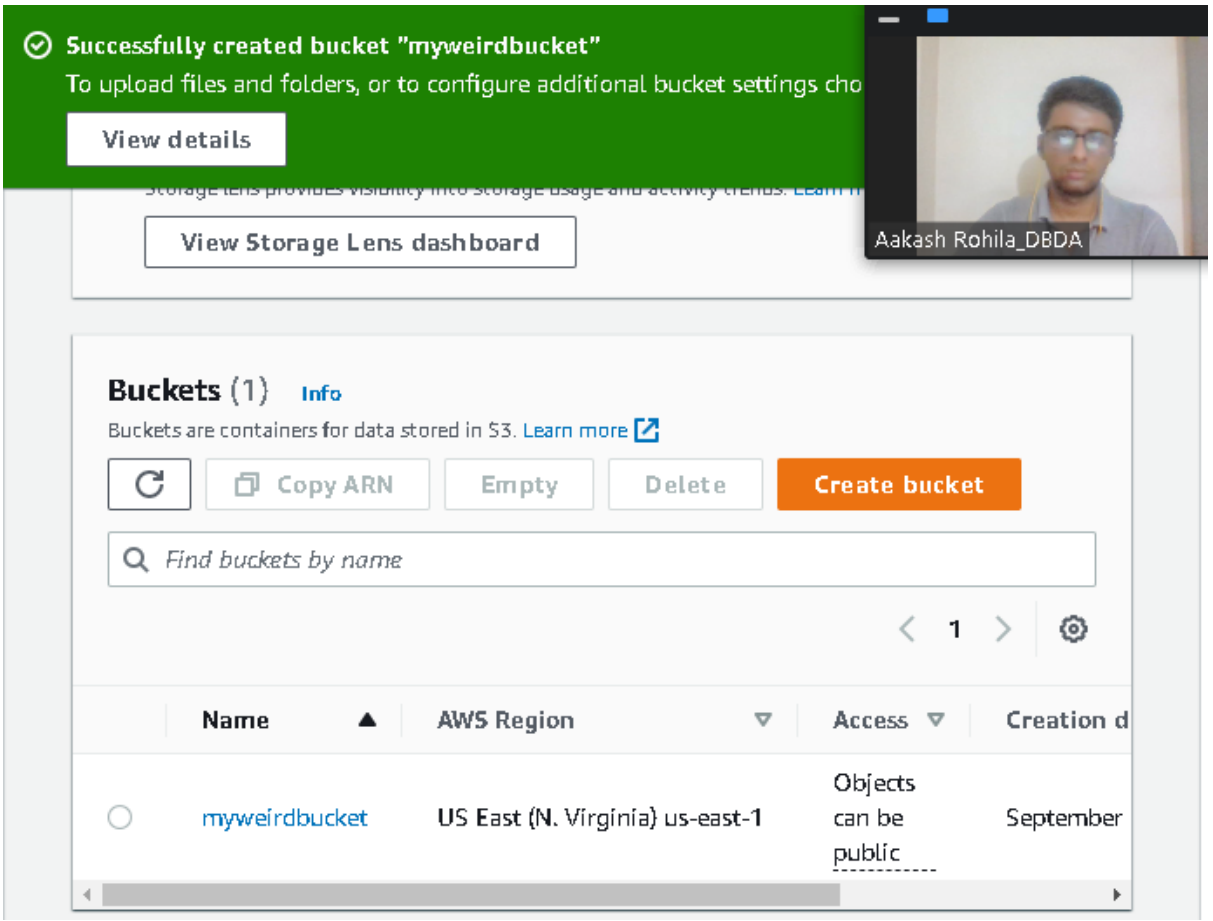
Step4>>Make ACL Enabled and make public traffic accessible…



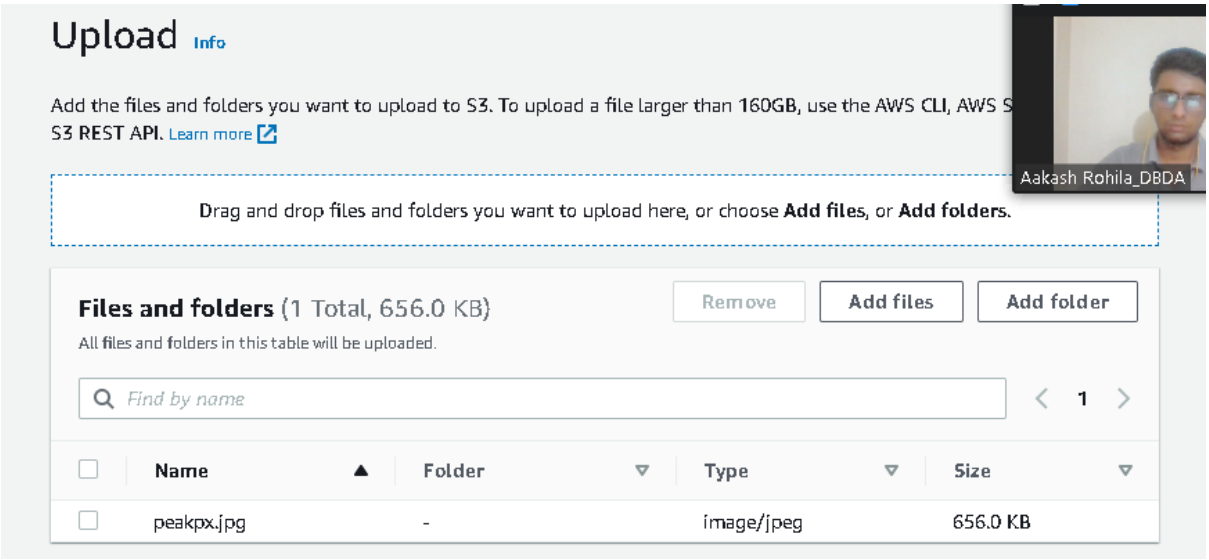
Step5>>Click on create bucket



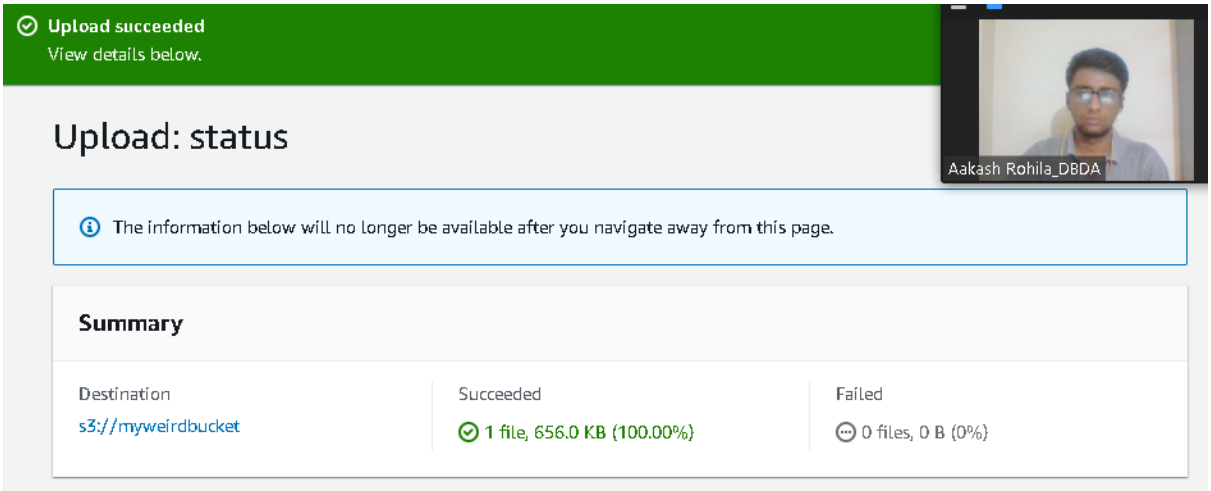
Step6>>And it should create your bucket successfully



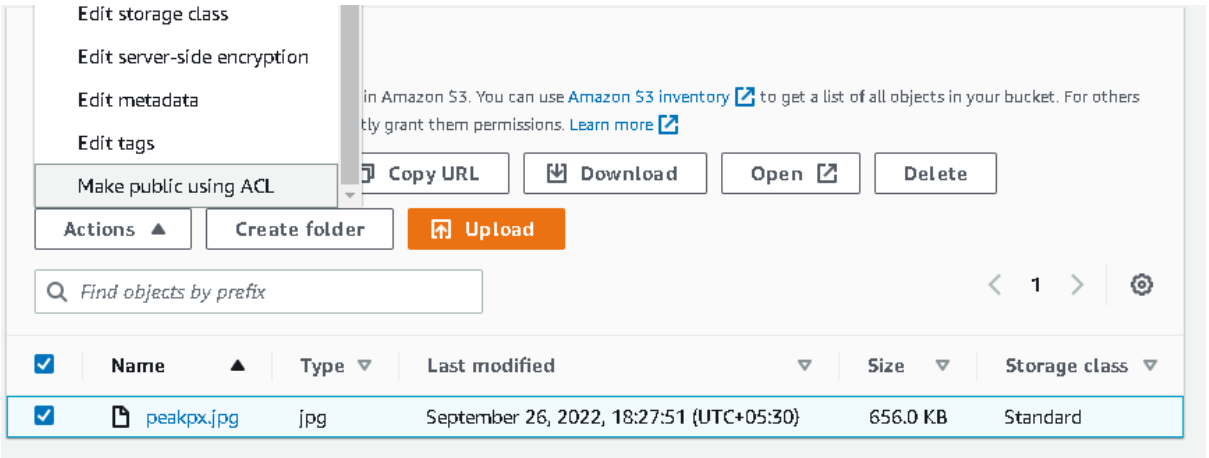
Step7>>To upload a file click on your bucket name link and click on upload file.



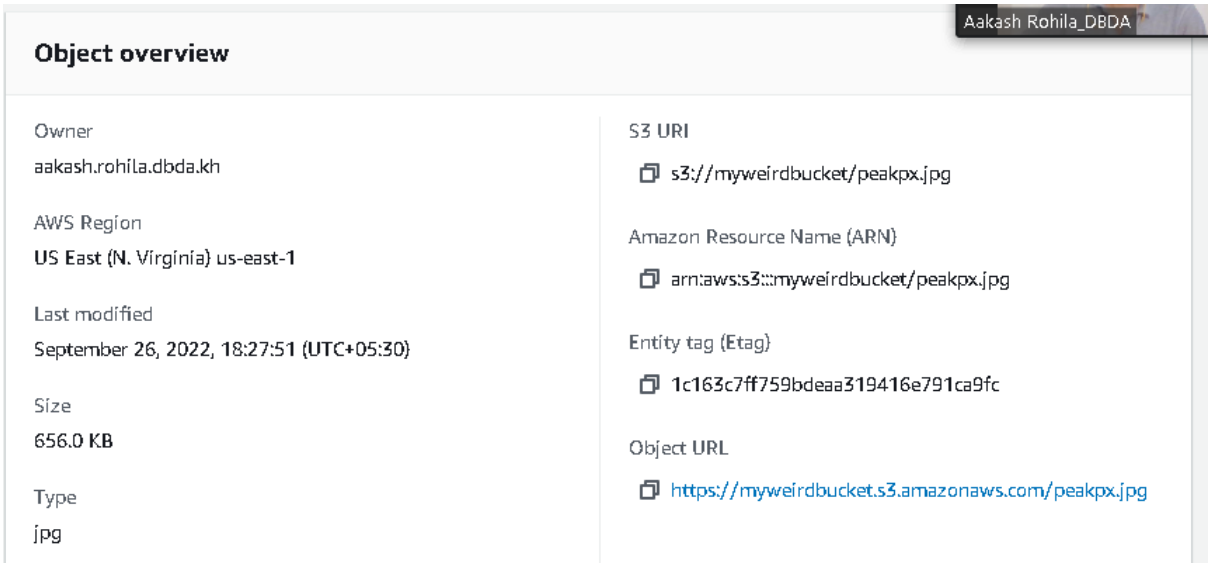
Step8>>It will show upload successfully



Step9>>Now in action centre make your file accessible i.e. Make public using ACL.



Step10>>Through the object link now we can access the file:



Step11>>Now this image is accessible through public ip…