

# In this Hand-on we are going to see benefits of using AMI (Amazon machine image).

## Let me give you one scenario-

Suppose your manager asked you to deploy company's application in North Virginia or in any other region.

You have successfully done all the setup which took you let's say 3-4 hours.

Now he want you to do the same thing in different region as per client requirements.

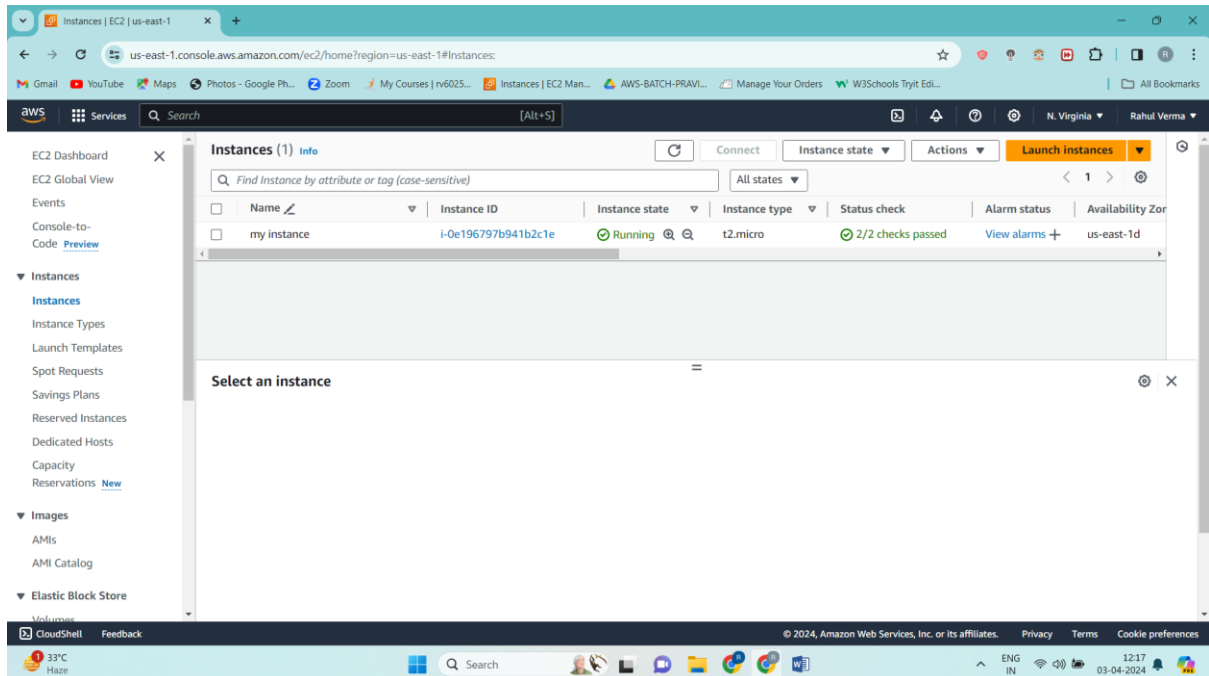
So if you do this manually then again it's going to take another 3-4 hours of yours.

So to overcome this thing we can use AMI, It'll copy all the data present in your Instance. And you can create an instance in other region using this AMI which will save your time.

## Prerequisite:

1) You have to create an Ec2 instance- I'm using Ubuntu os.

I have created this instance in US east North Virginia region.

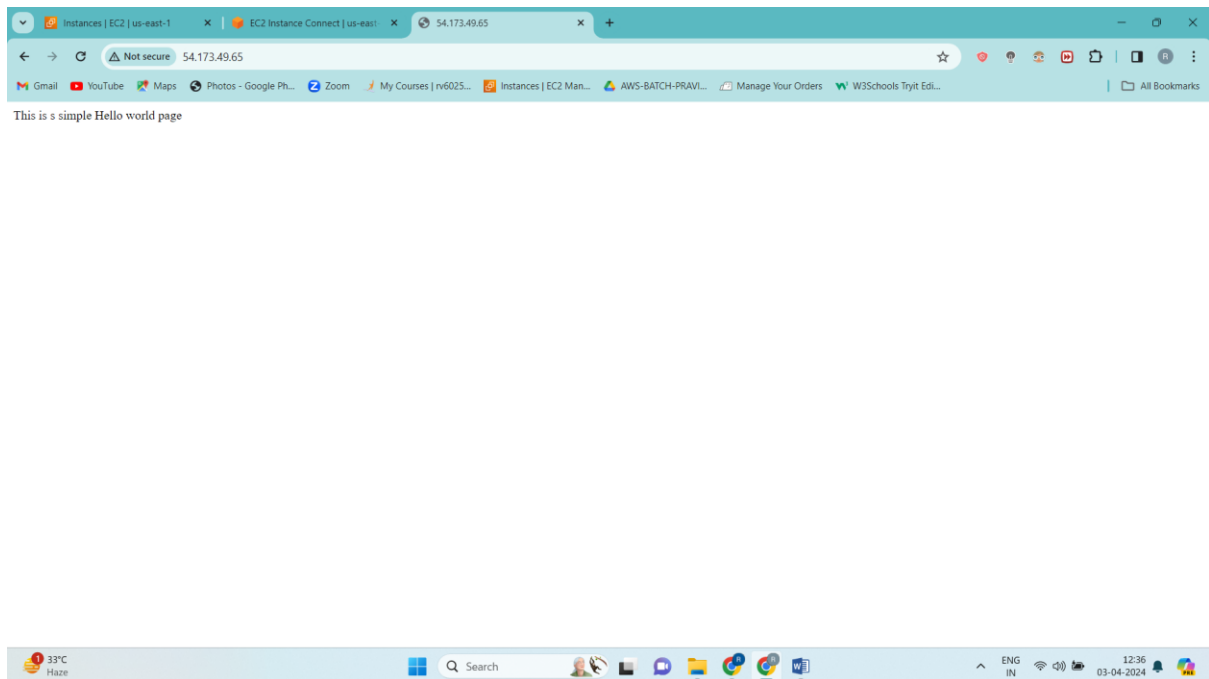


Commands to install web page in it-

```
aws | Services | Search
root@ip-172-31-29-79:/var/www/html# history
 1 apt-get update
 2 apt-get install apache2 -y
 3 cd /var/www/html
 4 ls
 5 rm -r index.html
 6 nano index.html
 7 clear
 8 history
root@ip-172-31-29-79:/var/www/html#
```

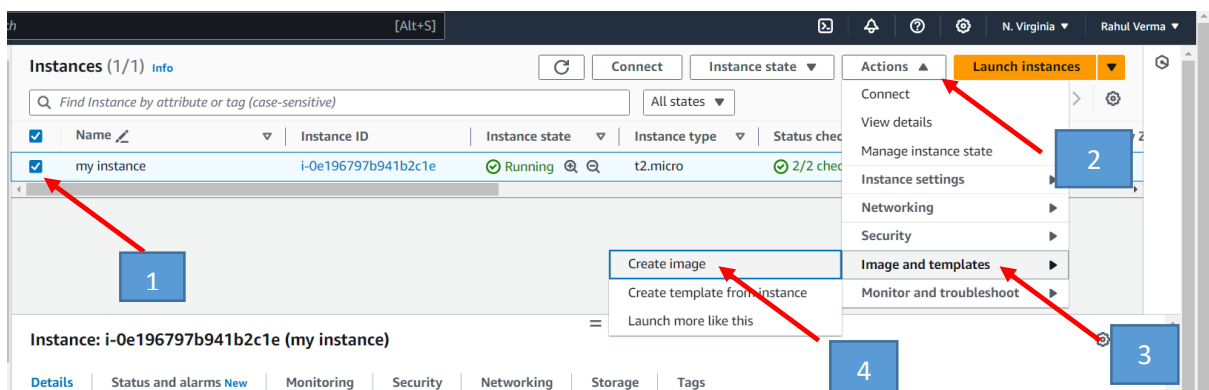
After writing nano index.html write anything you want I just wrote  
**“This is a simple Hello world page”**

If you copy paste your instance's public IP address in your browser, it will show like this.

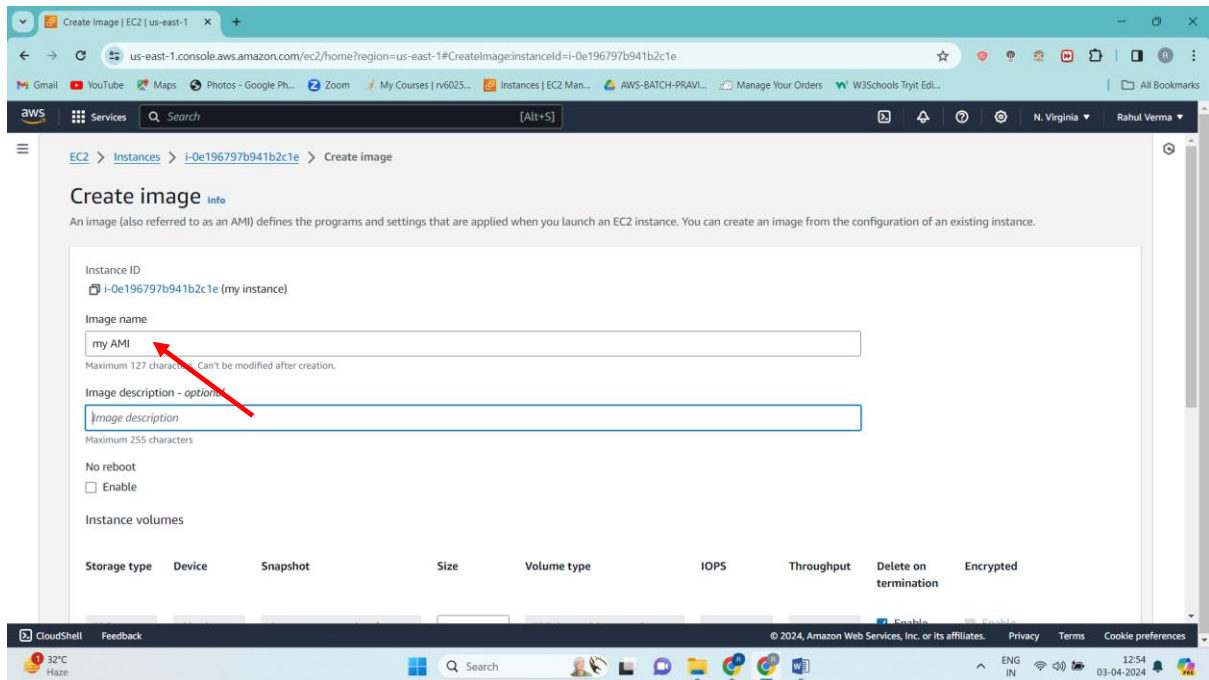


**Step 1:** Now let's start with our Hands-on

- 1) To create AMI select your instance
- 2) Now go to **Actions** under that,
- 3) Click on Image and templates and then,
- 4) Click on **Create Image**

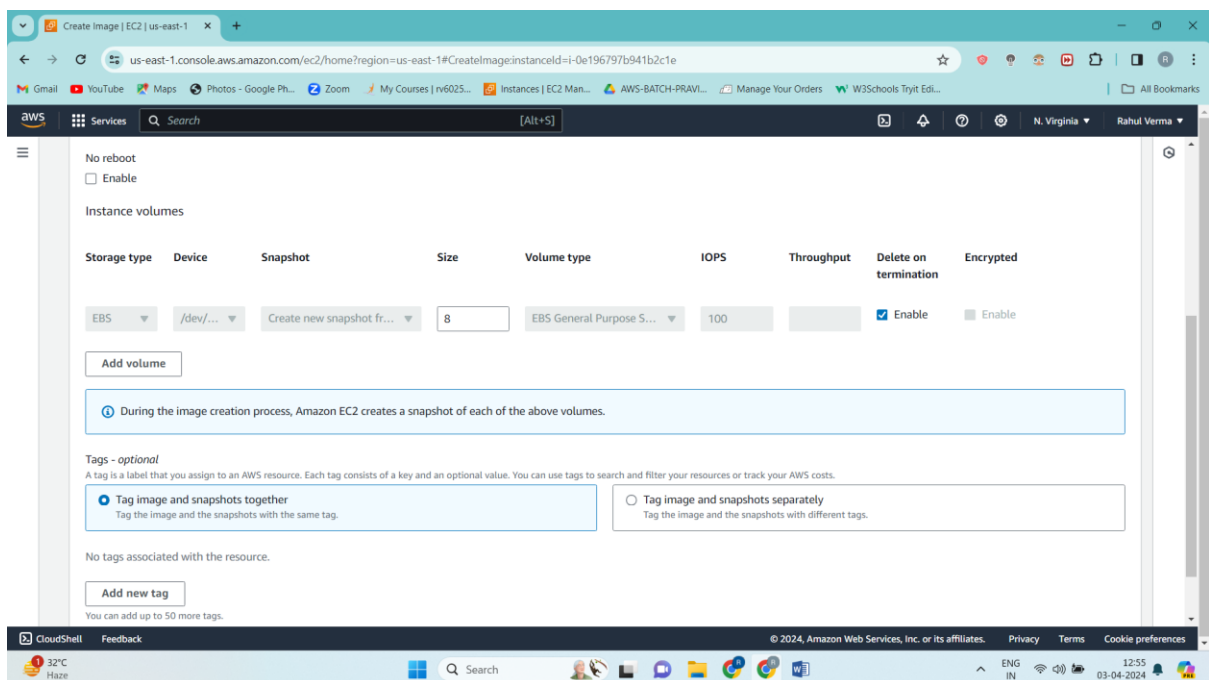


## Step 2: Define image name

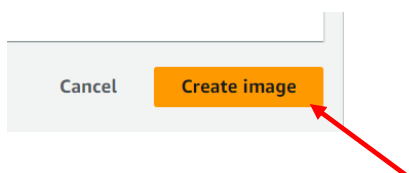


The screenshot shows the AWS Management Console 'Create image' page. The page title is 'Create image' with an 'Info' icon. Below the title, a description states: 'An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.' The 'Instance ID' is listed as 'i-0e196797b941b2c1e (my instance)'. The 'Image name' field is highlighted with a red arrow and contains the text 'my AMI'. Below it, the 'Image description - optional' field is empty. The 'No reboot' checkbox is unchecked. The 'Instance volumes' section is visible, showing a table with columns: Storage type, Device, Snapshot, Size, Volume type, IOPS, Throughput, Delete on termination, and Encrypted. The table currently has one row with 'EBS' as the storage type, '/dev/...' as the device, 'Create new snapshot fr...' as the snapshot, '8' as the size, 'EBS General Purpose S...' as the volume type, '100' as IOPS, and 'Delete on termination' set to 'Enable'.

You can add volume to it- I'm going with default settings and click on create image button

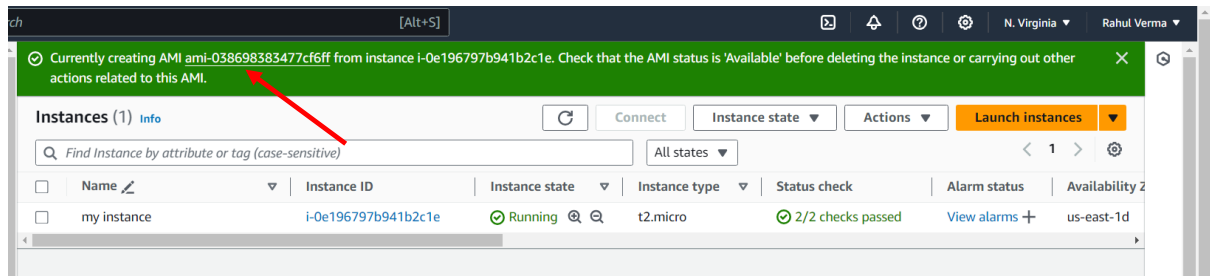


This screenshot shows the 'Instance volumes' section of the 'Create image' page. It displays a table with columns: Storage type, Device, Snapshot, Size, Volume type, IOPS, Throughput, Delete on termination, and Encrypted. The table has one row with 'EBS' as the storage type, '/dev/...' as the device, 'Create new snapshot fr...' as the snapshot, '8' as the size, 'EBS General Purpose S...' as the volume type, '100' as IOPS, and 'Delete on termination' set to 'Enable'. Below the table, there is an 'Add volume' button. A message box states: 'During the image creation process, Amazon EC2 creates a snapshot of each of the above volumes.' The 'Tags - optional' section is also visible, with two radio buttons: 'Tag image and snapshots together' (selected) and 'Tag image and snapshots separately'. The 'Add new tag' button is at the bottom.

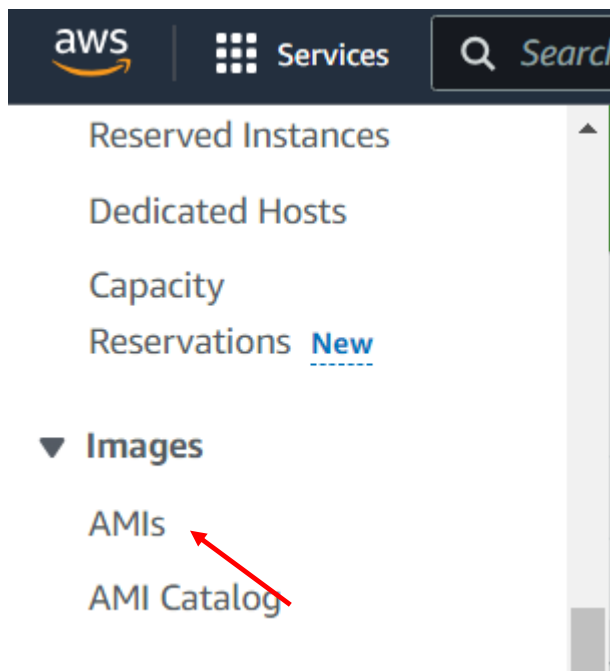


This screenshot shows the bottom of the 'Create image' page. It features a 'Cancel' button and a 'Create image' button. A red arrow points to the 'Create image' button.

Our image is created now



**Step 3:** on you left hand side scroll down and click on Images under images click on AMIs

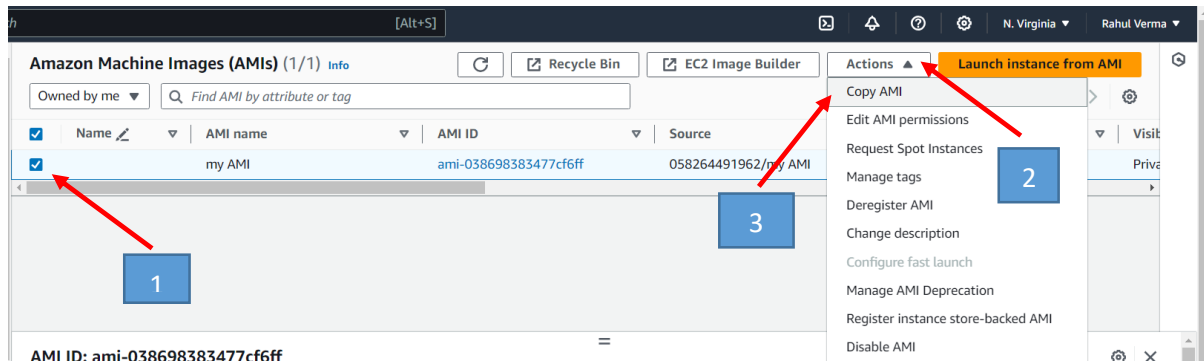


**Step 4:** to create instance using AMI in another region

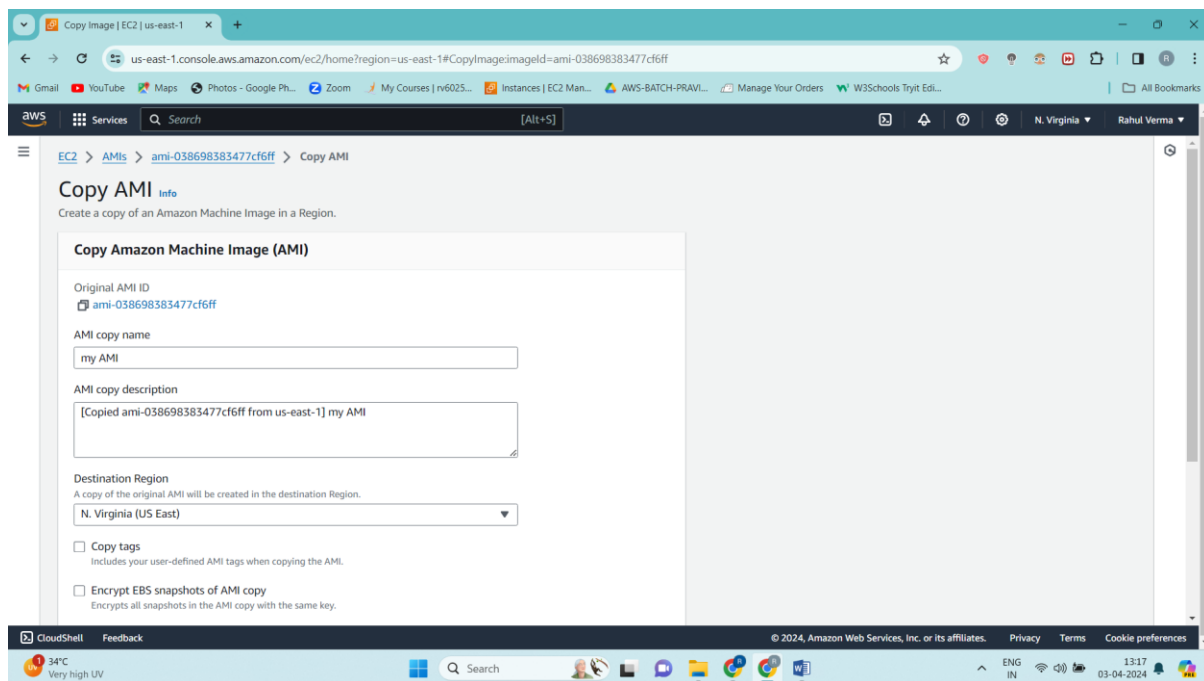
1) Select you AMI

2) Go to Actions

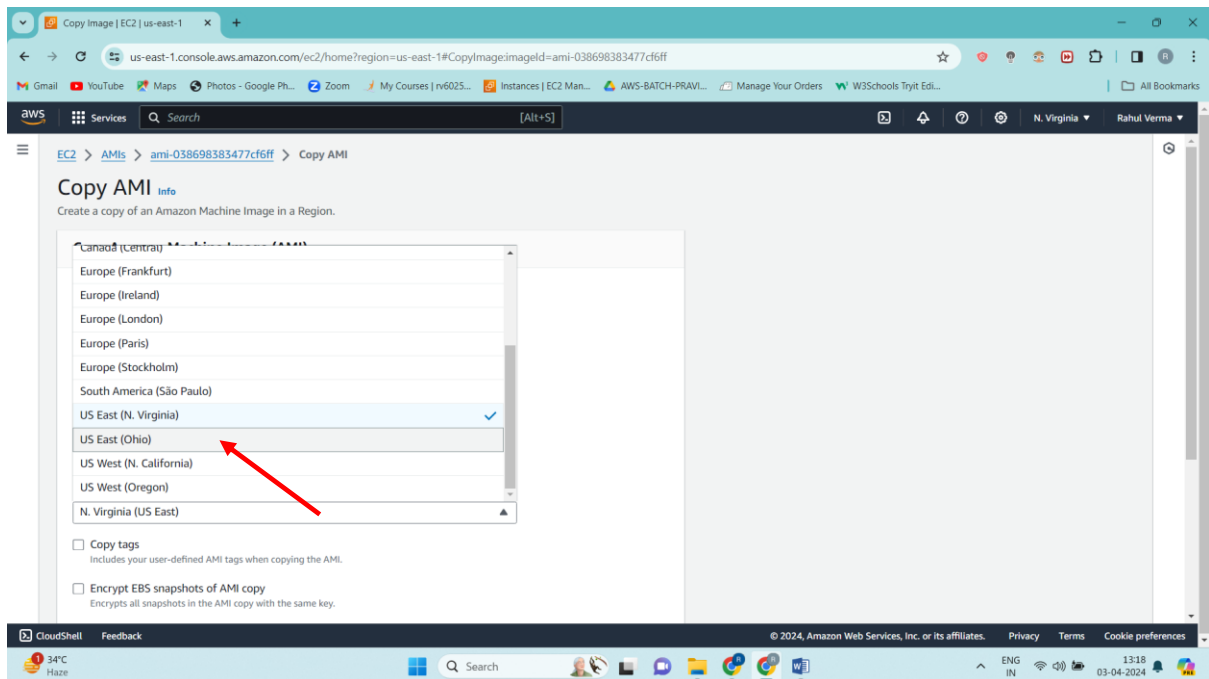
3) Click on Copy AMI



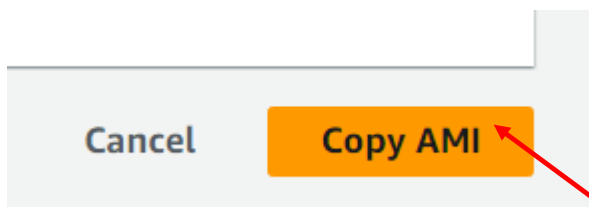
This is how it will appear after clicking on Copy AMI



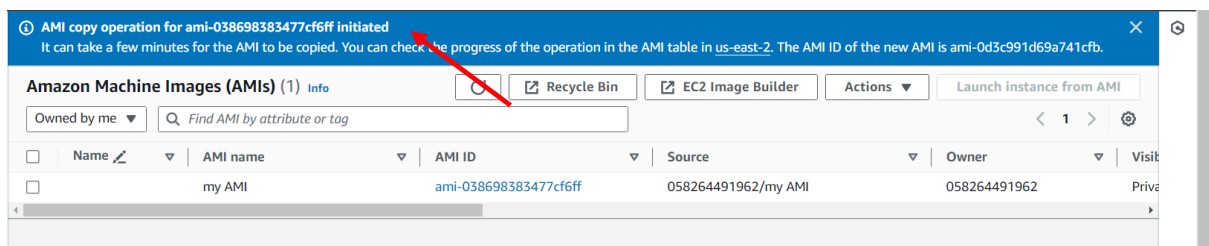
**Step 5:** Now select the region where you want to create this instance. I'm selecting Us east Ohio region.



And click on copy image



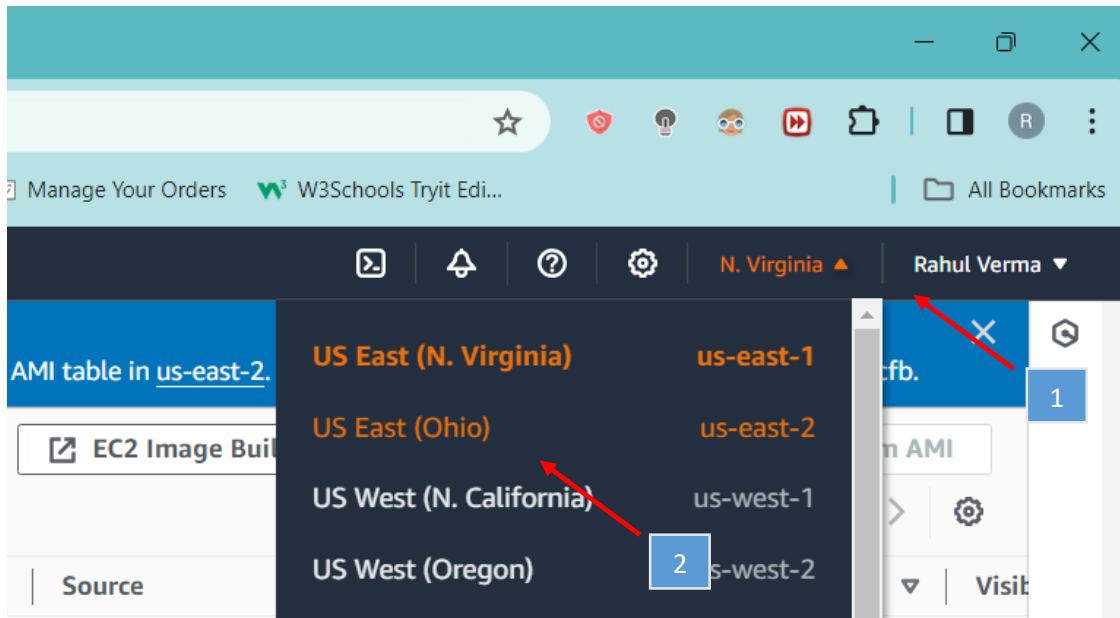
After clicking on copy AMI it will show you like this



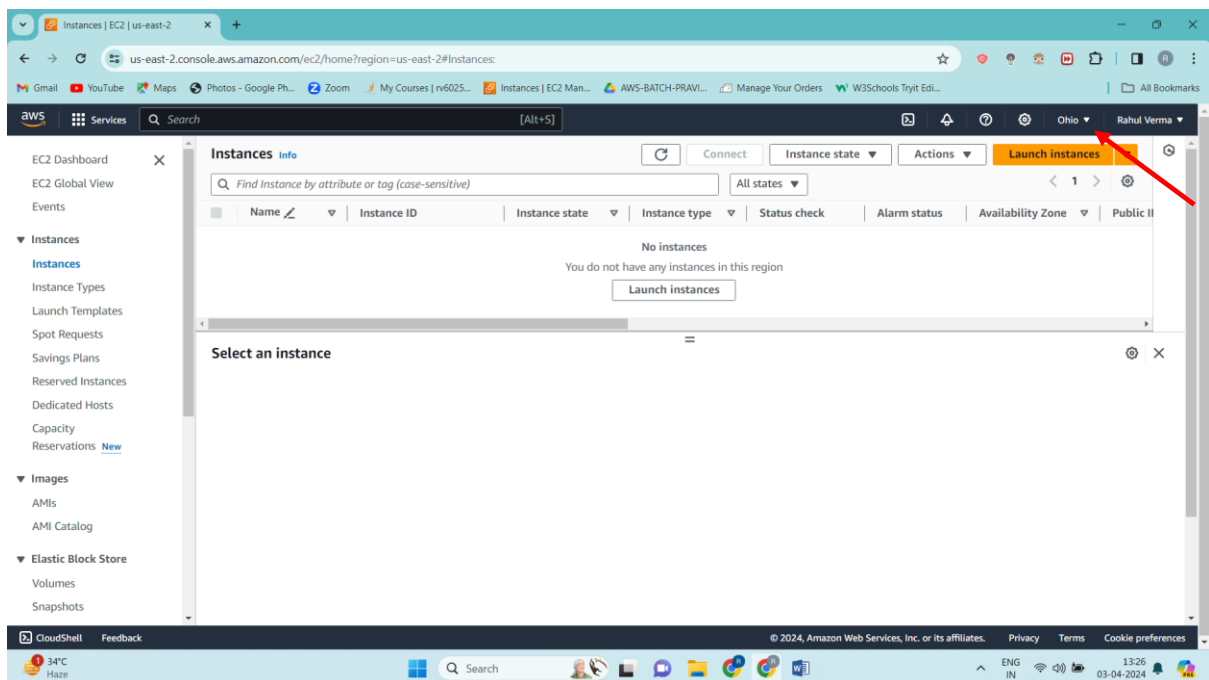
**Step 6:** Now we have to switch to Ohio region

1) For that just click on the current region

2) Now select the desired region where you copied your AMI. In my case I copied it to Ohio so im going to select Ohio only.

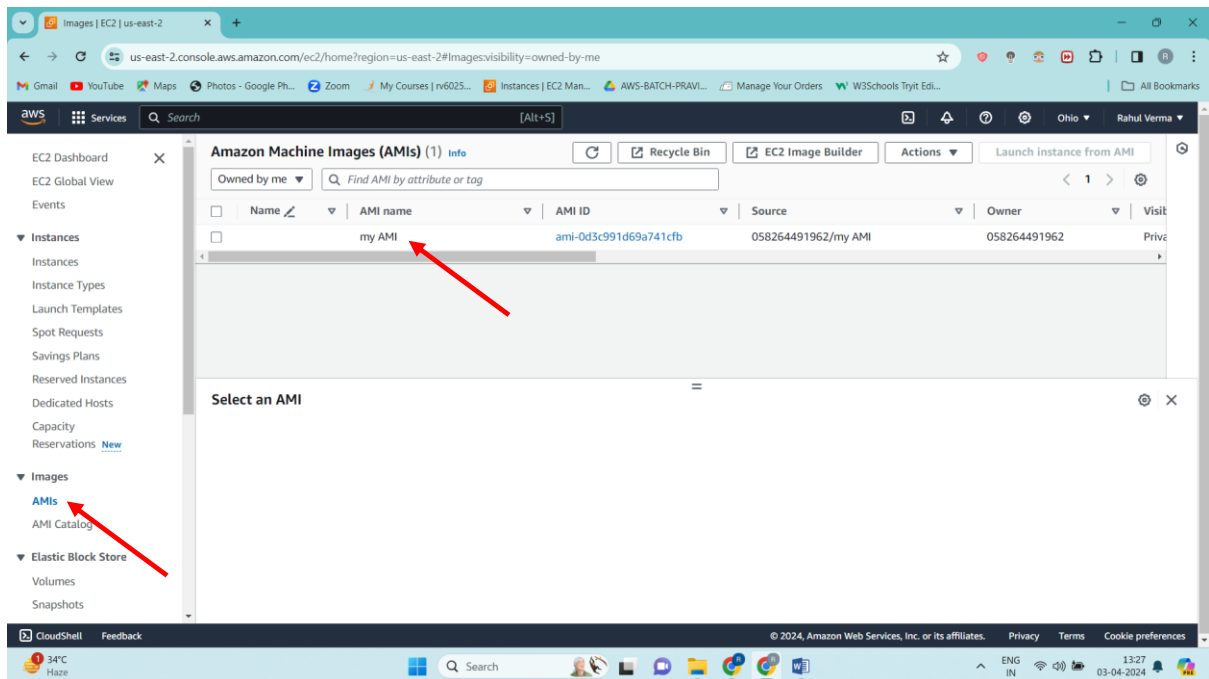


And now you can see in the below pic that we have switched to Ohio region. And it doesn't have any instance in it.





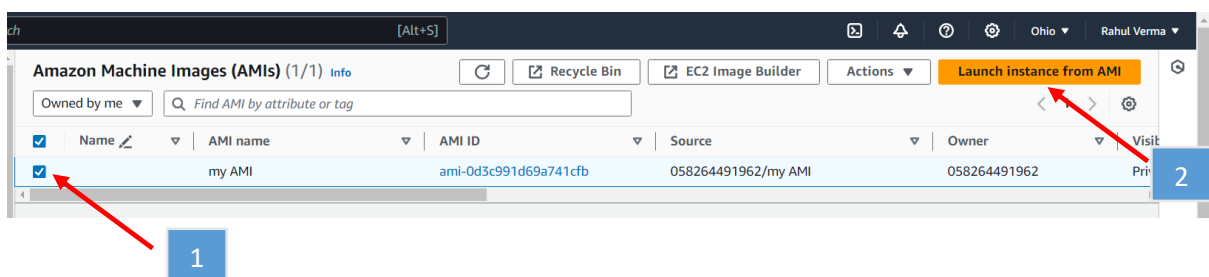
**Step 7:** now let's check our AMI. And you can see our AMI is there in AMIs section.



**Step 8:** Final step to create instance

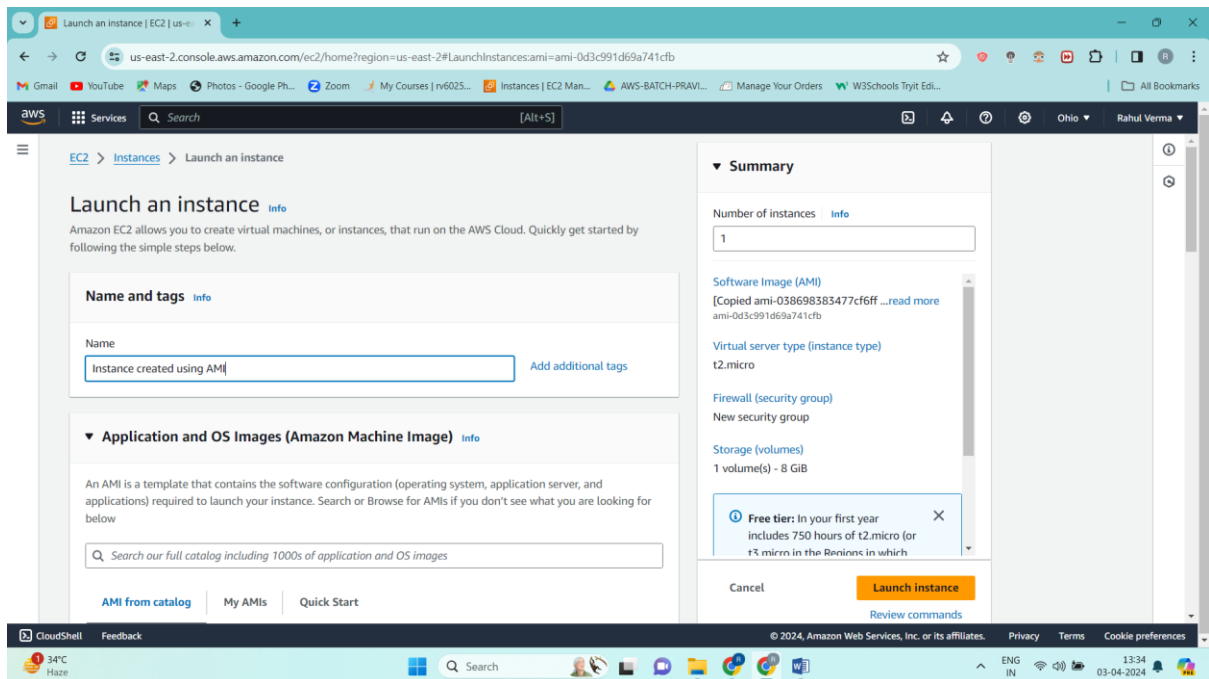
1) Select you AMI

2) Now just click on **Launch instance from AMI**

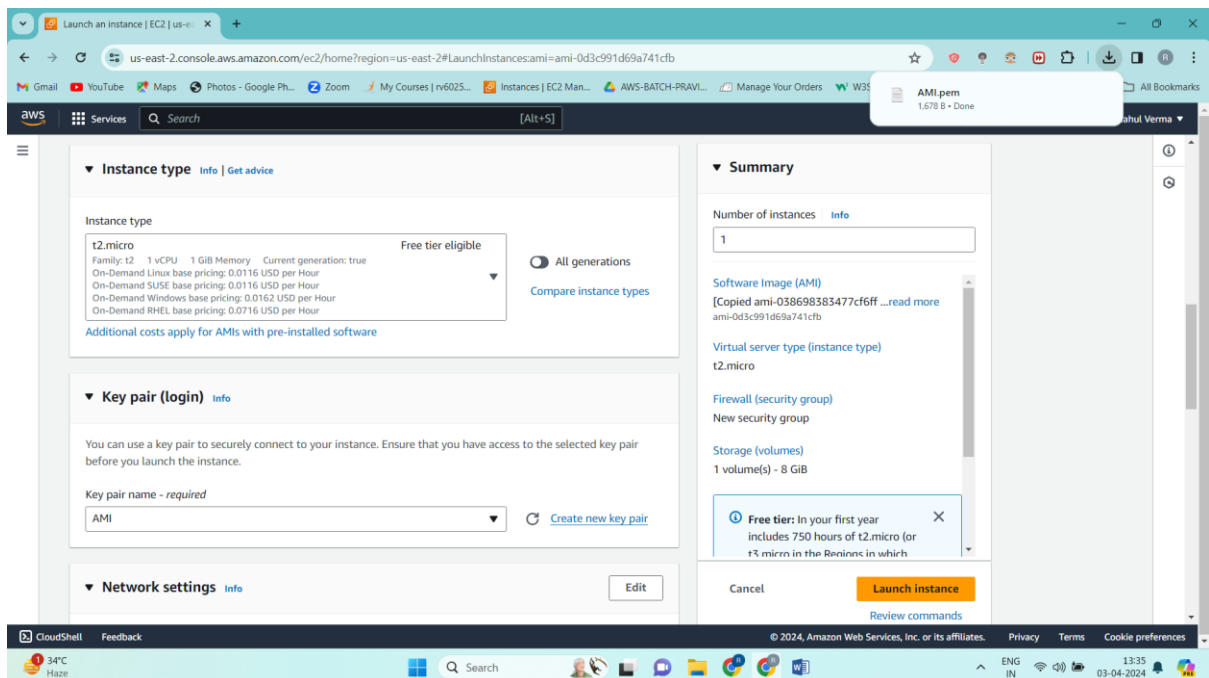


Now you just have to create your instance

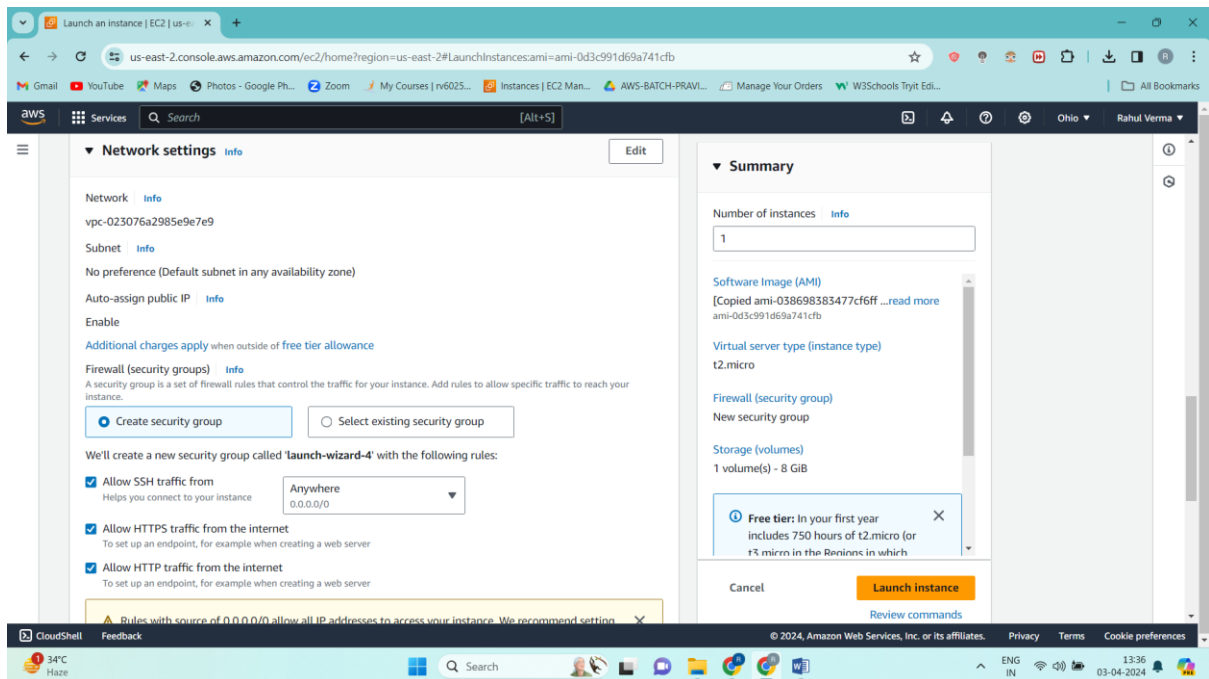
Define it's name



Select instance type and select key pair if you have any or else create one new key pair for this instance

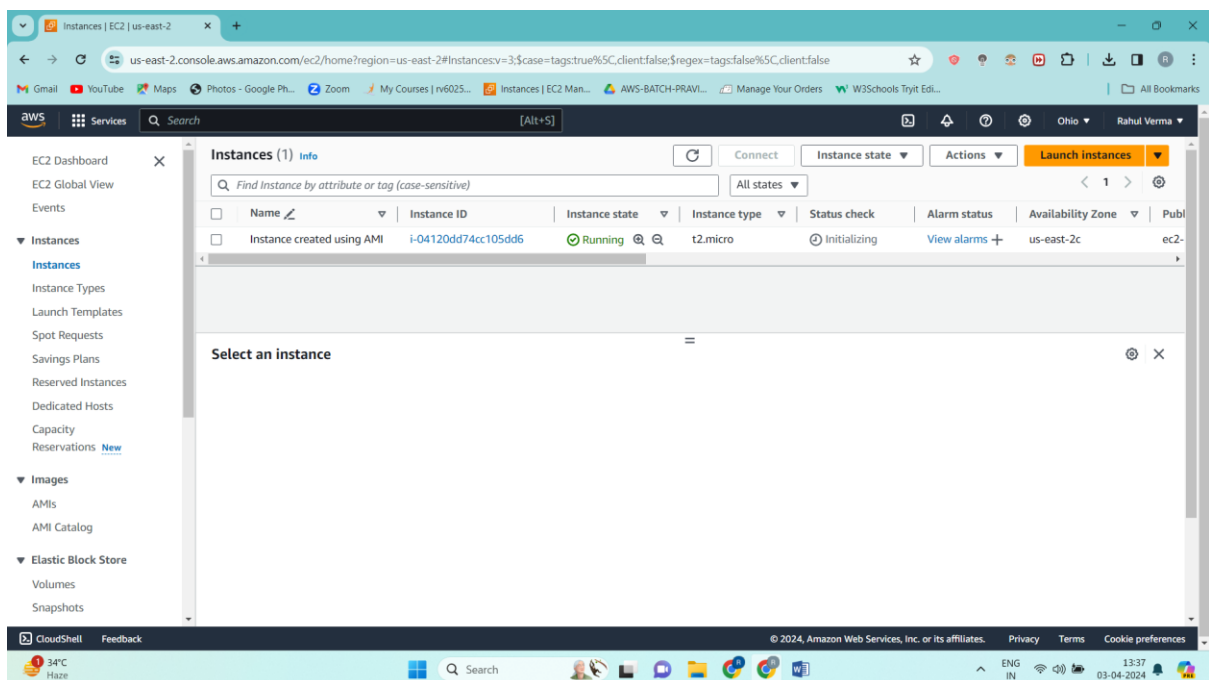


Select security group or create new one and just click on Launch instance

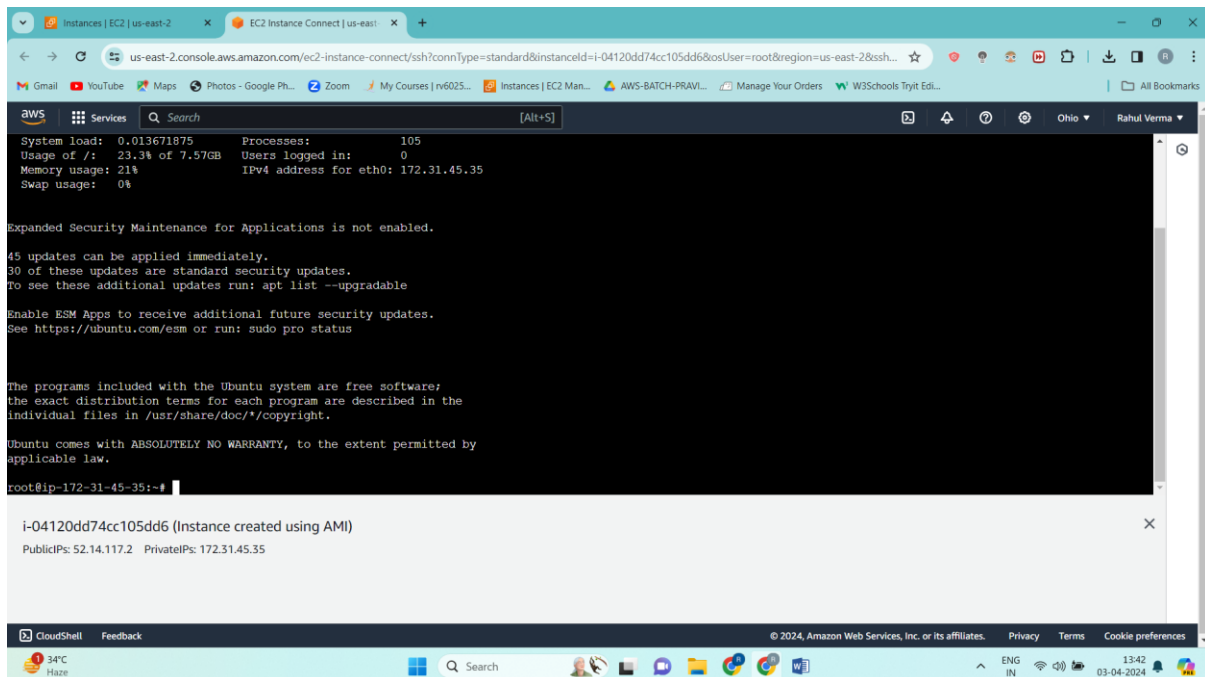


Our instance is created now

And you can check it's status is in initializing state as well as it's in ohio region only.



**Step 9:** Now we let's check if our index.html file is there or not so for that let's connect to this newly created instance.



The screenshot shows the AWS CloudShell interface. The terminal window displays system information for an Ubuntu instance, including system load, processes, memory usage, and network details. It also shows security updates and a warning about expanded security maintenance. Below the terminal, a terminal window for instance i-04120dd74cc105dd6 is open, showing its public and private IP addresses.

```
System load: 0.013671875   Processes:      105
Usage of /:  23.3% of 7.57GB   Users logged in: 0
Memory usage: 21%           IPv4 address for eth0: 172.31.45.35
Swap usage:  0%

Expanded Security Maintenance for Applications is not enabled.

45 updates can be applied immediately.
30 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

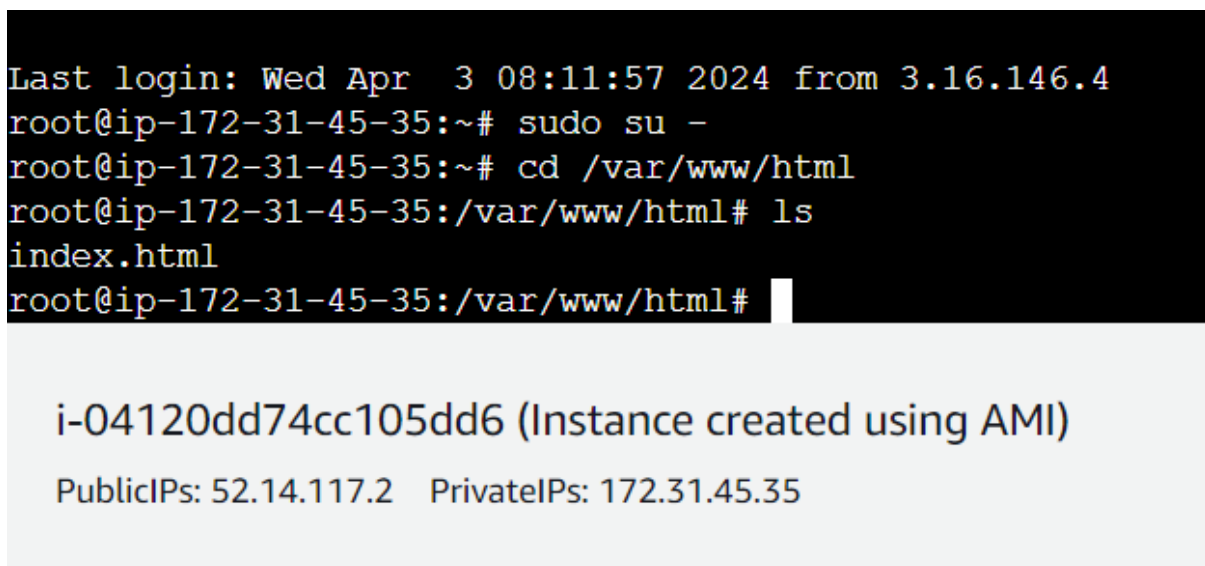
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

root@ip-172-31-45-35:~#
```

i-04120dd74cc105dd6 (Instance created using AMI)  
PublicIPs: 52.14.117.2 PrivateIPs: 172.31.45.35

And you can see our file is there

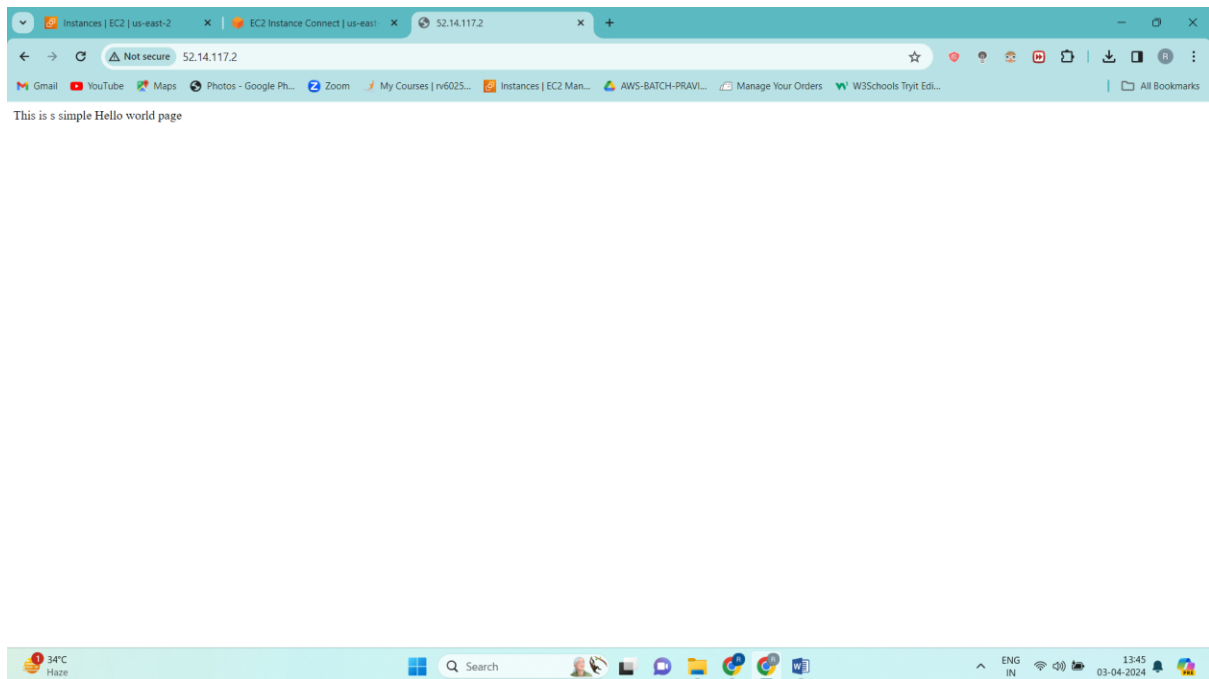


The screenshot shows the AWS CloudShell terminal window. The user has logged in and is now in the root user environment. They have navigated to the /var/www/html directory and listed the files, showing that index.html is present.

```
Last login: Wed Apr  3 08:11:57 2024 from 3.16.146.4
root@ip-172-31-45-35:~# sudo su -
root@ip-172-31-45-35:~# cd /var/www/html
root@ip-172-31-45-35:/var/www/html# ls
index.html
root@ip-172-31-45-35:/var/www/html#
```

i-04120dd74cc105dd6 (Instance created using AMI)  
PublicIPs: 52.14.117.2 PrivateIPs: 172.31.45.35

Let's verify if our web page is working or not will just copy paste it's IP address in browser



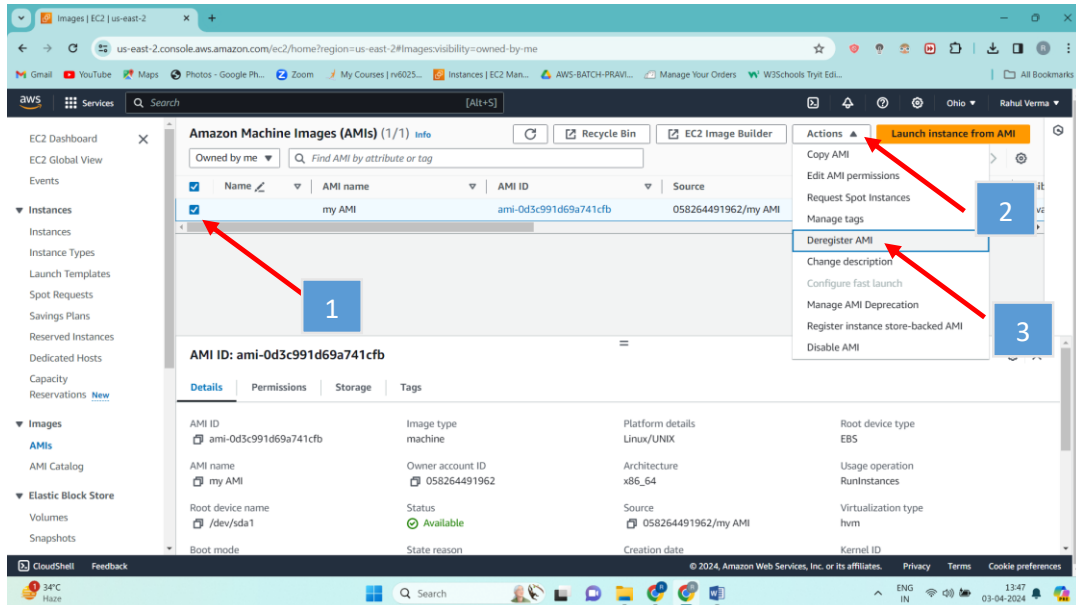
And it's working good.

Now after AMI is copied or once your new instance is created in ohio region. If you delete index.html file or if you terminate your instance from North Virginia region, then also your web page will be there in your new instance which is created using AMI in ohio region.

**Note: Delete all the resources after use to avoid charges**

## To delete AMI.

- 1) Select your AMI
- 2) Go to actions and
- 3) Click on Deregister AMI



# Thank You