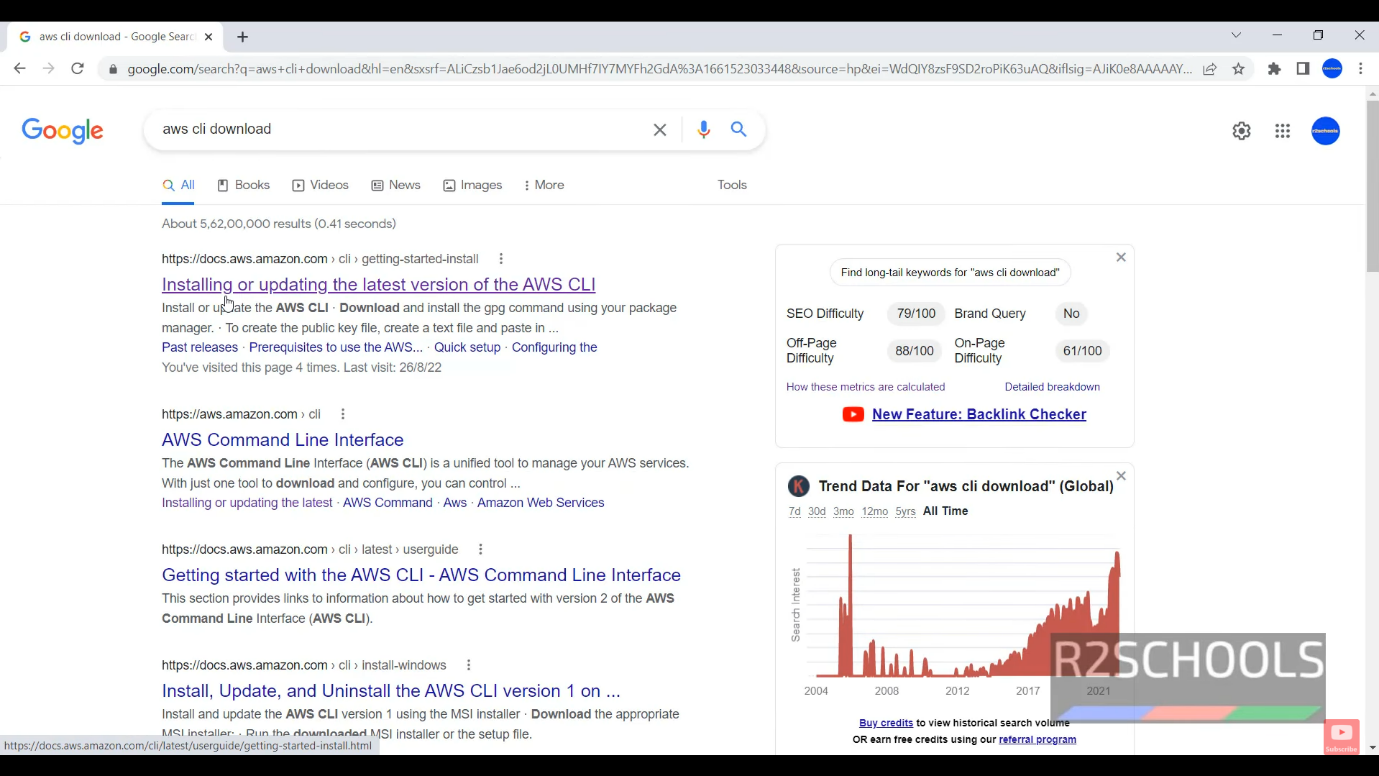
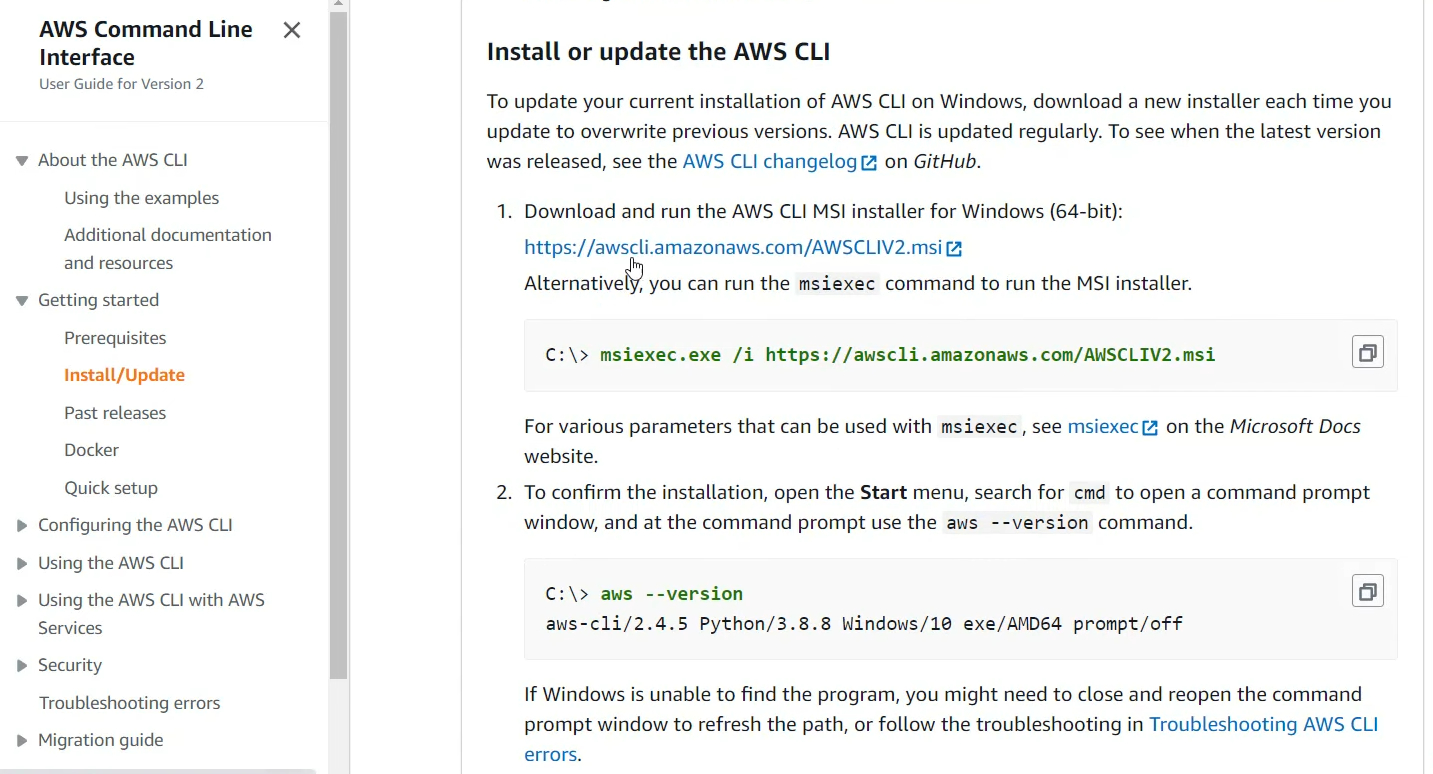
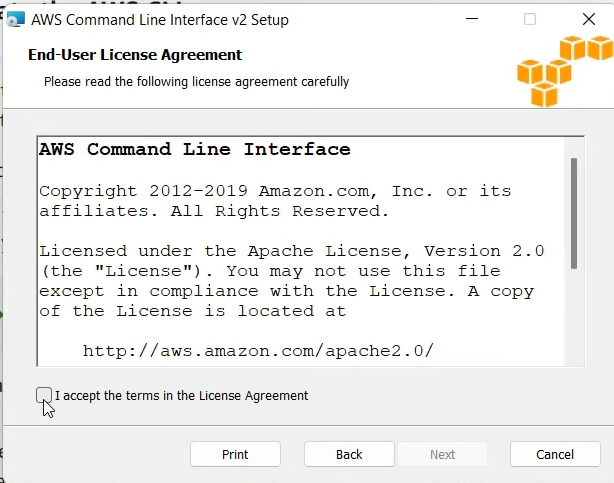
**Step 1: Install and configure aws CLI**

Search for **aws cli download** and click on the link

****

Click on the link for the **msi(Micro Star International)** **file** the download process of the msi file will start automatically

****

as you click on the **installed msi** file the following will get displayed click on **accept** the terms in the License Agreement

A screenshot of a computer

Description automatically generatedthen cick on next  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
then click on **install**

A screenshot of a computer

Description automatically generated

finally it will start installing **aws cli** on the local machine

A screenshot of a computer

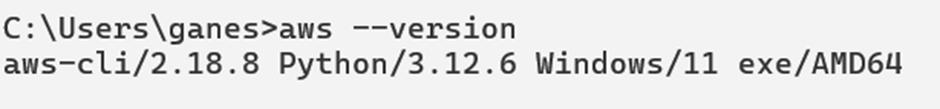
Description automatically generated

After installation is finished click on finish

A screenshot of a computer

Description automatically generated

Go to local machine cmd and type command **aws –version** the output should be   
**aws-cli/2.18.8 Python/3.12.6 Windows/11 exe/AMD64** the versions can be different

****

In order to **configure aws cli** there are **2 methods** to do so **:**go on the aws academy site, provided by the college but remember it has only the student access and student doesn’t has the access to create new user in IAM if you try to create it the following error will pop up

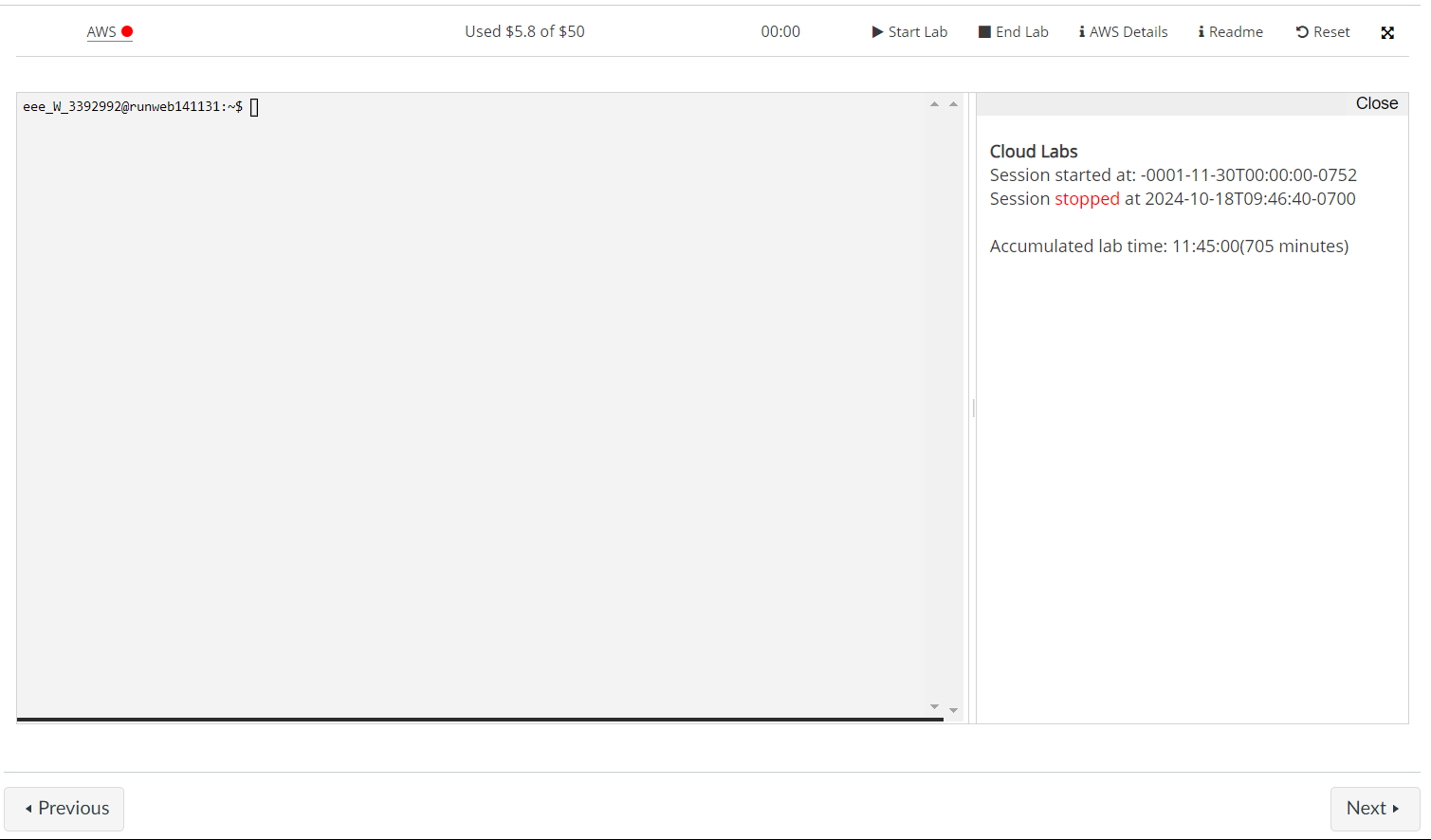
**Method 1(create new user in IAM):**

****

**The second method(configure using temporary credentials, note it will be different for each lab session):**

**a)This is how you can get the credentials**

**when the lab is not started there are no credentials allotted to the user**

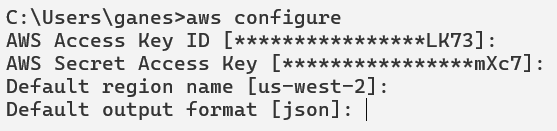


**b)as we can see there no credentials available here so in order to get the credentials we have to click on start the lab**

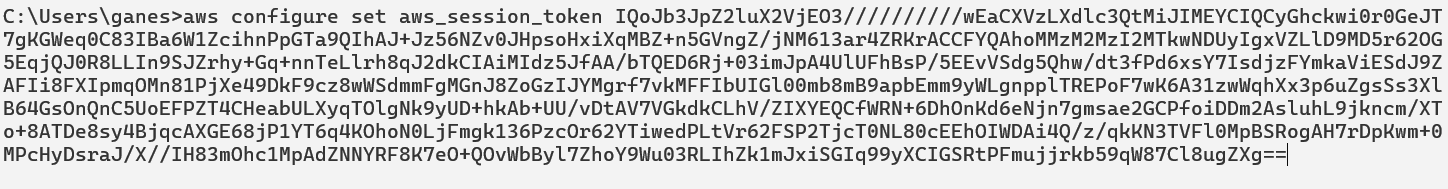
**A screenshot of a computer

Description automatically generated**

**c)Go to cmd and type the command “aws configure” and enter the details from the aws cli details above**

****

**d)As we know this is session details which is temporary we also have to enter the additional command “aws configure set aws\_session\_token <Your\_Session\_Token>”**

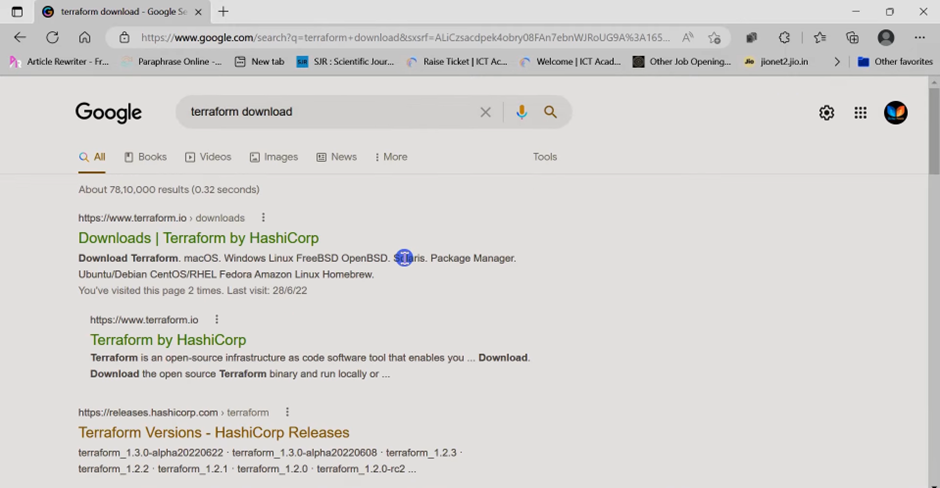
****

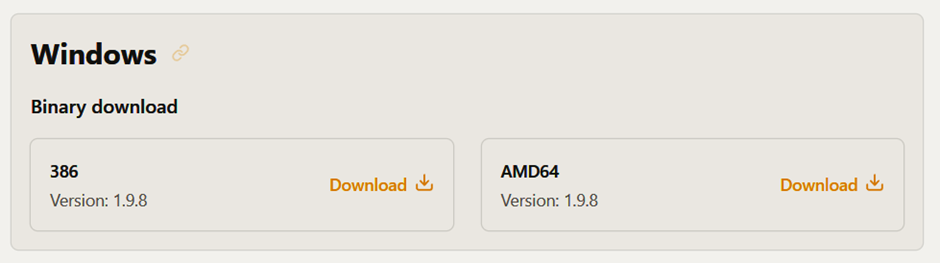
**Cli will finaly get configured to our aws academy account**

**And you can confirm that by using command “aws ec2 list-instances”**

Step 2: **Install and configure Terraform**

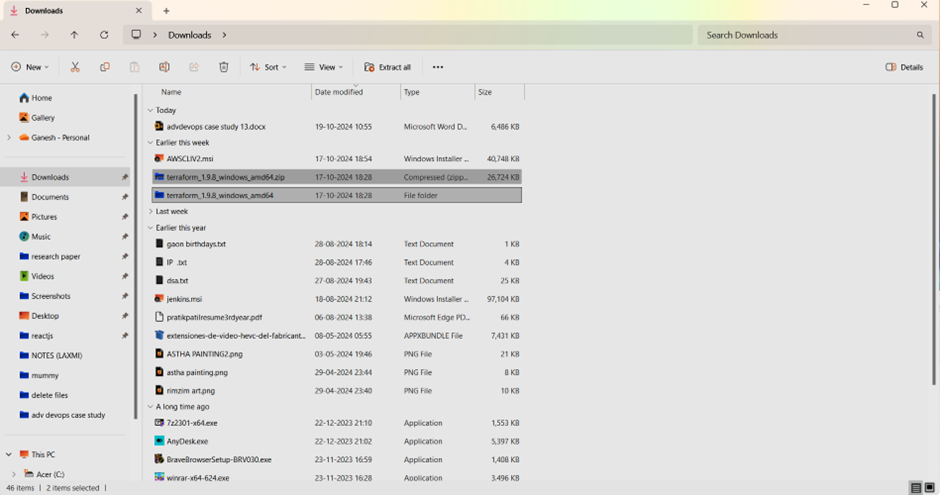
Download the zip file of the terraform for the windows from the official website of the hashicorp

****

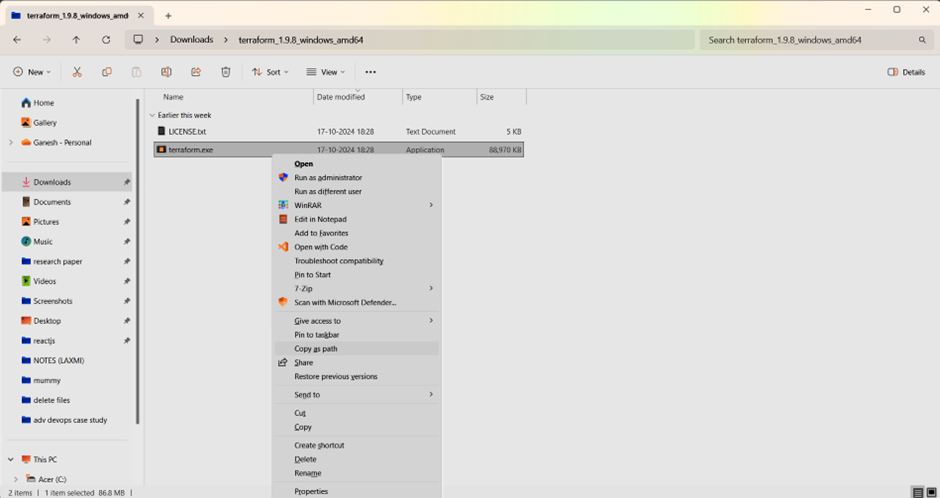
****

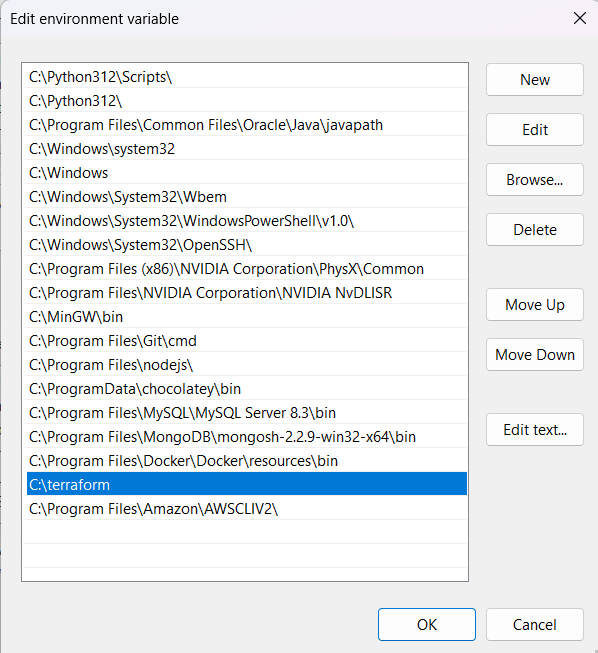
in my case I have downloaded the amd64 you can download any on them whichever supports your machine

after downloading it you have to **unzip** it



Navigate inside of the unzip file and **copy** **it’s path**

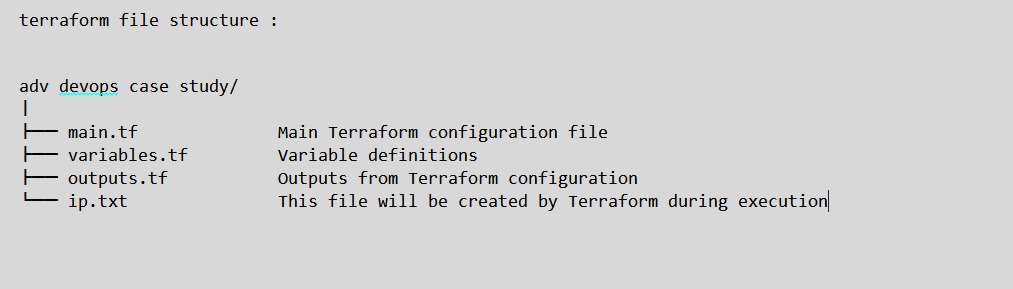


And add that path to environment variables

Check the terraform through cmd using **terraform –version**



Step 3: **write the terraform code**

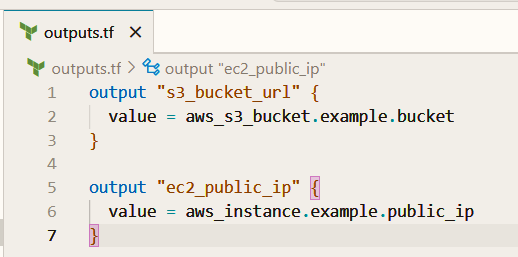
make the following directory structure in that new working folder for ex: adv devops case study 



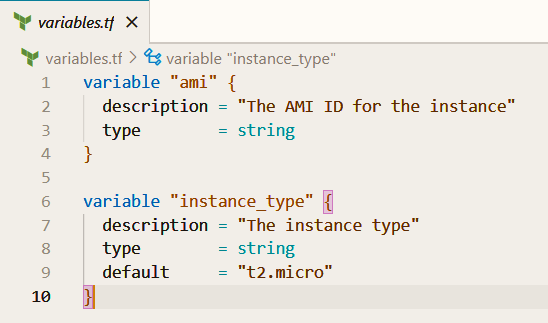
A screenshot of a computer program

Description automatically generated**main.tf** terraform code :

output.tf code :

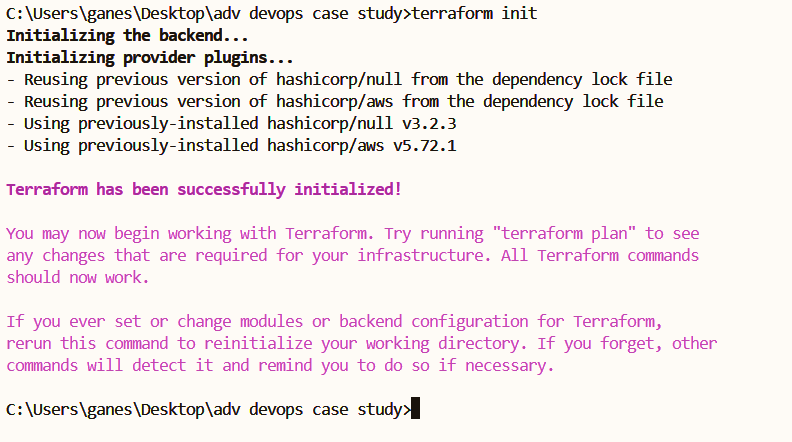


variables.tf



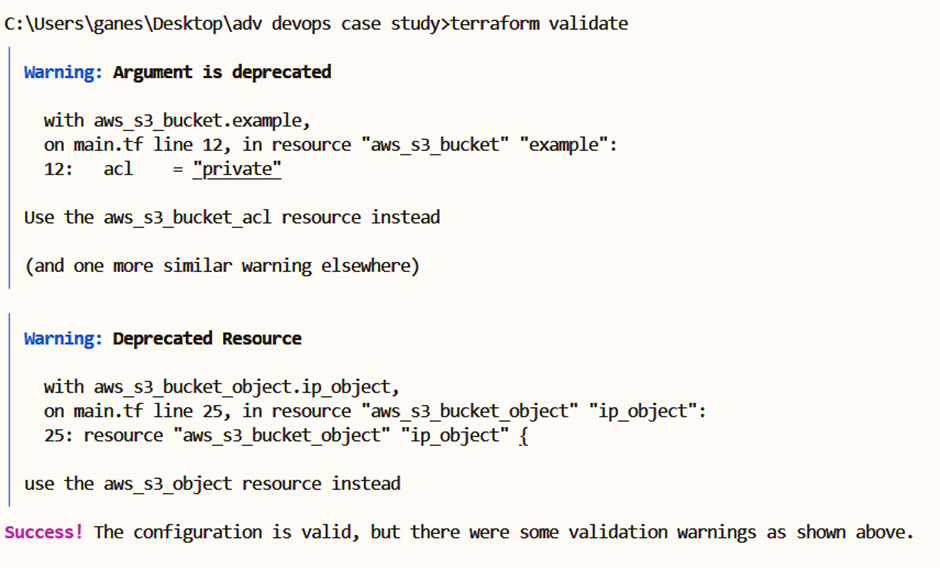
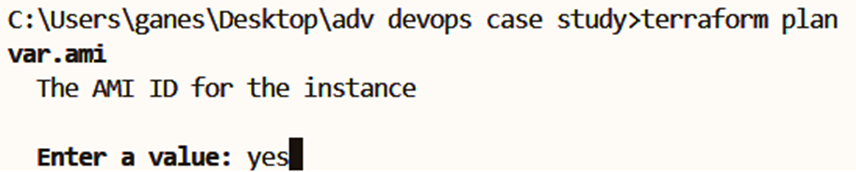
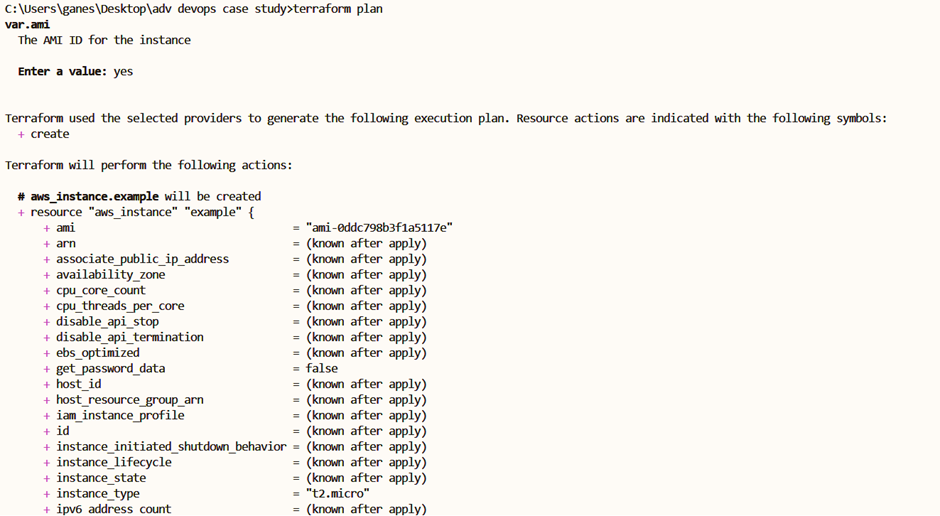
Step 4: **run terraform commands**

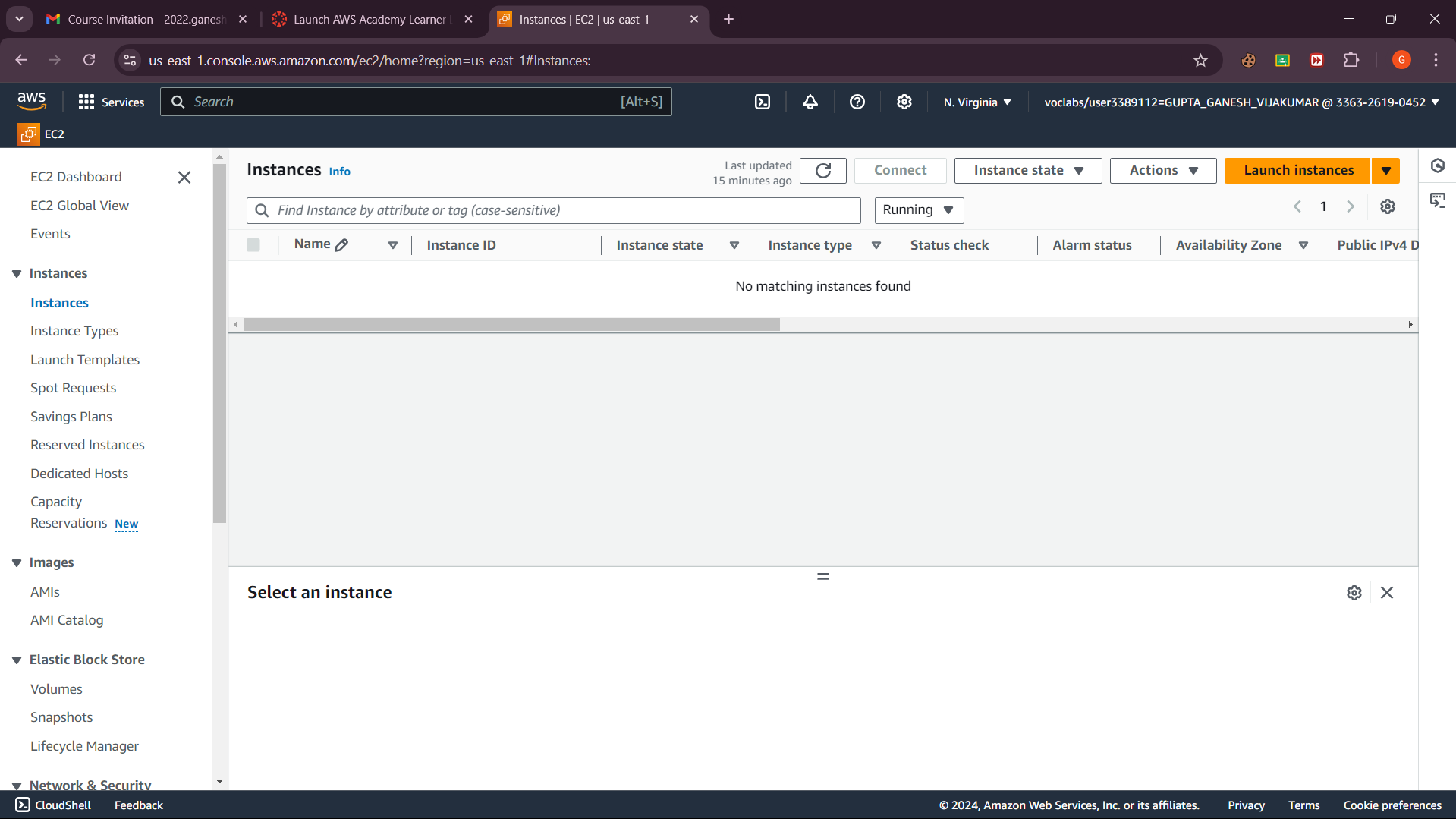
1. **“terraform init” command**

****

1. **” terraform fmt” command:**

****

1. **“terraform validate” command:  
   **
2. **“terraform plan” command:   
     
   enter yes and enter to see the plan that is going to execute on aws   
   **

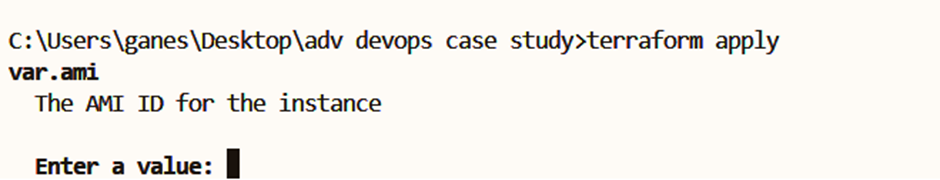
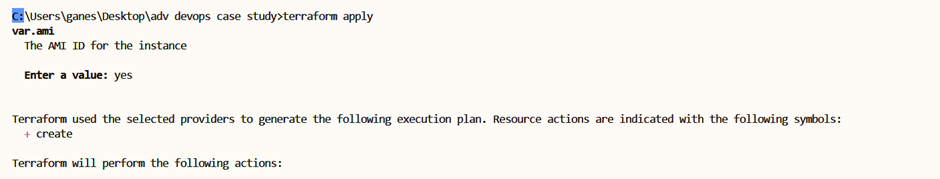
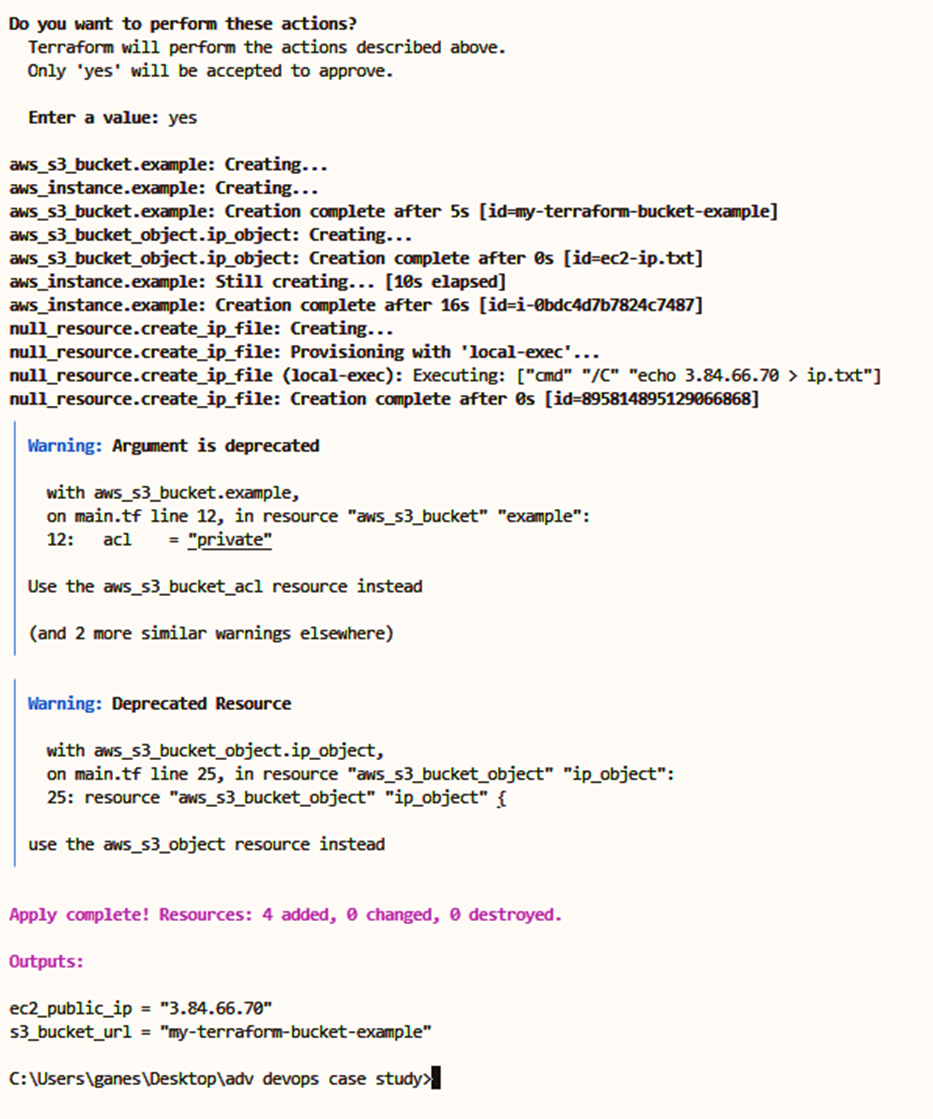
**Before executing “terraform apply” command there is no ec2 instance running  
**

**Before executing “terraform apply” command there are only 2 s3 bucket instances and note that every s3 bucket has unique name**

**A screenshot of a computer

Description automatically generated**

1. **Executing “terraform apply” command**

**Now check the aws ec2 instances and s3 bucket at academy**

1 ec2 instance is created A screenshot of a computer

Description automatically generated

1 s3 bucket is also created

A screenshot of a computer

Description automatically generated

**Now click on s3 bucket that is newly created   
A screenshot of a computer

Description automatically generated**

**Now click on ec2-ip.txt**

**A screenshot of a computer

Description automatically generated**

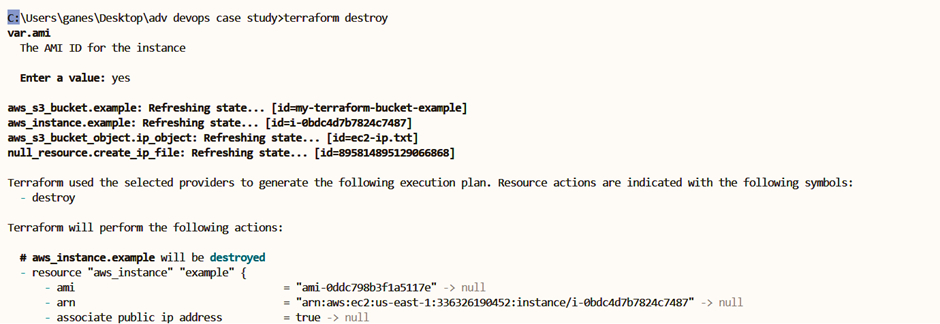
**Click on download**

**A screenshot of a computer

Description automatically generated**

**Open it **

**The ip address is stored inside the s3 bucket is finally proved**

**And then use terraform destroy command to destroy all the instances that are made**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**EC2 instance and S3 bucket is destroyed.**