**TERRAFORM**

**AWS – cloud formation**

**AZURE – arm template**

**GOOGLE CLOUD – deployment manager**

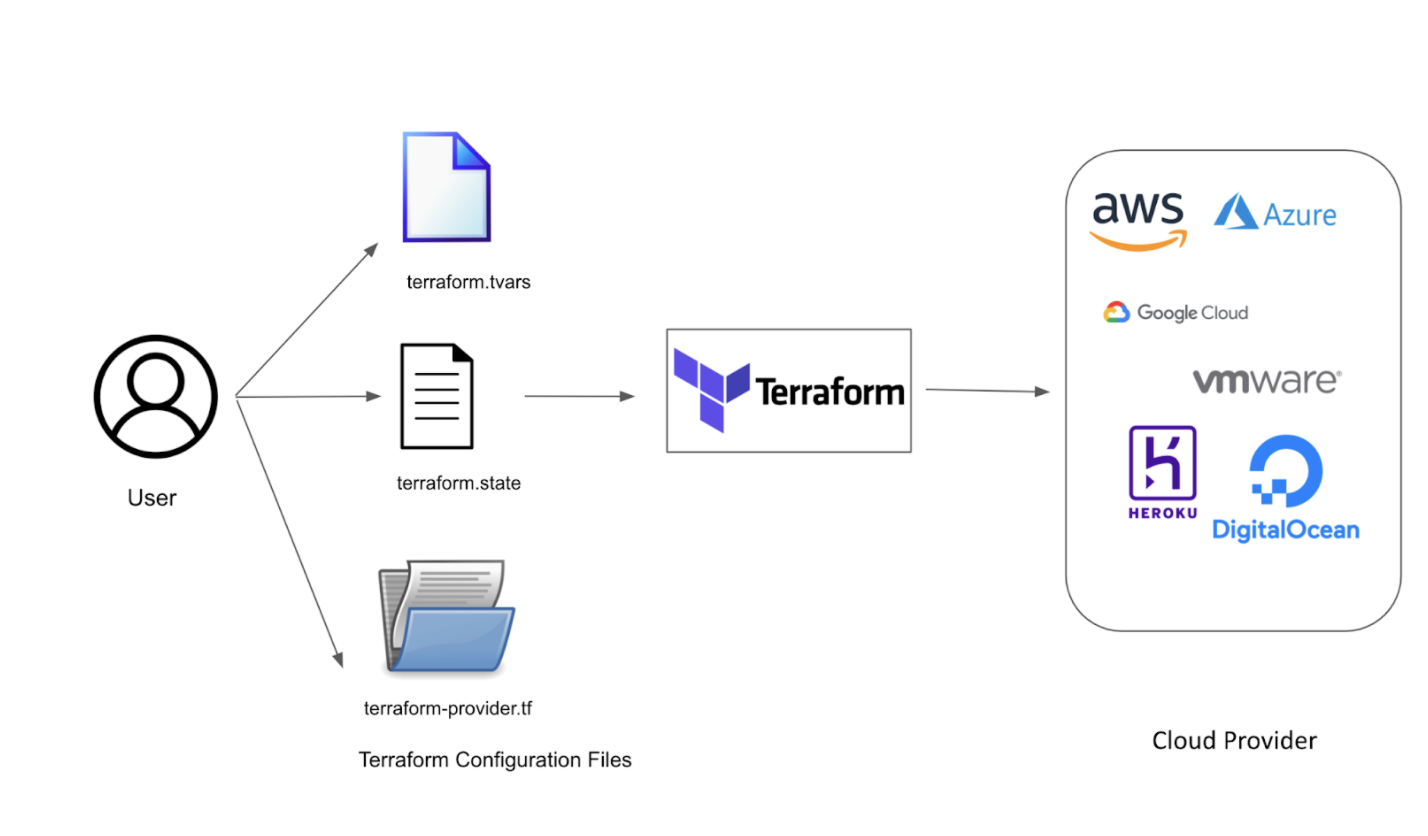
**But terraform do infrastructure automation on all**

**Terraform uses hasicorp configure lang (hcl)**

**IAC – manage and provisioning infra through code instead of through manual confi.**

**Imperatative – we need to define systematic way of code sequence.**

**Declarative – we just need to define desired state,iac will configure for you.**



**Providers : aws,azure,googe cloud**

**Terraform init : to initialize working directory**

**Terraform validate : whether configuration is syntactically correct**

**Terraform plan : to create execution environment plan,show what changes occur after apply but don’t do it actually.**

**Terraform apply : used to apply changes required to reach desired state.**

**Terraform destroy : it destroy terraform managed infrastructure.**

**TERRAFORM COMMANDS**

**HCL SYNTAX :**

**Blocks : it group expressions,arguments into a labelled structure,which external blocks can acess it.**

**Arguments : logical expressions in key-value pair.**

**Expressions : it either represent comput values such as string or numeric value.**

**<block type> “<block label>””<block label name>” {**

**<identifier> = <expression>**

**}**

**For giving aws provider access,store it in main.tf file**

**terraform {**

**required\_providers {**

**aws = {**

**source = "hashicorp/aws"**

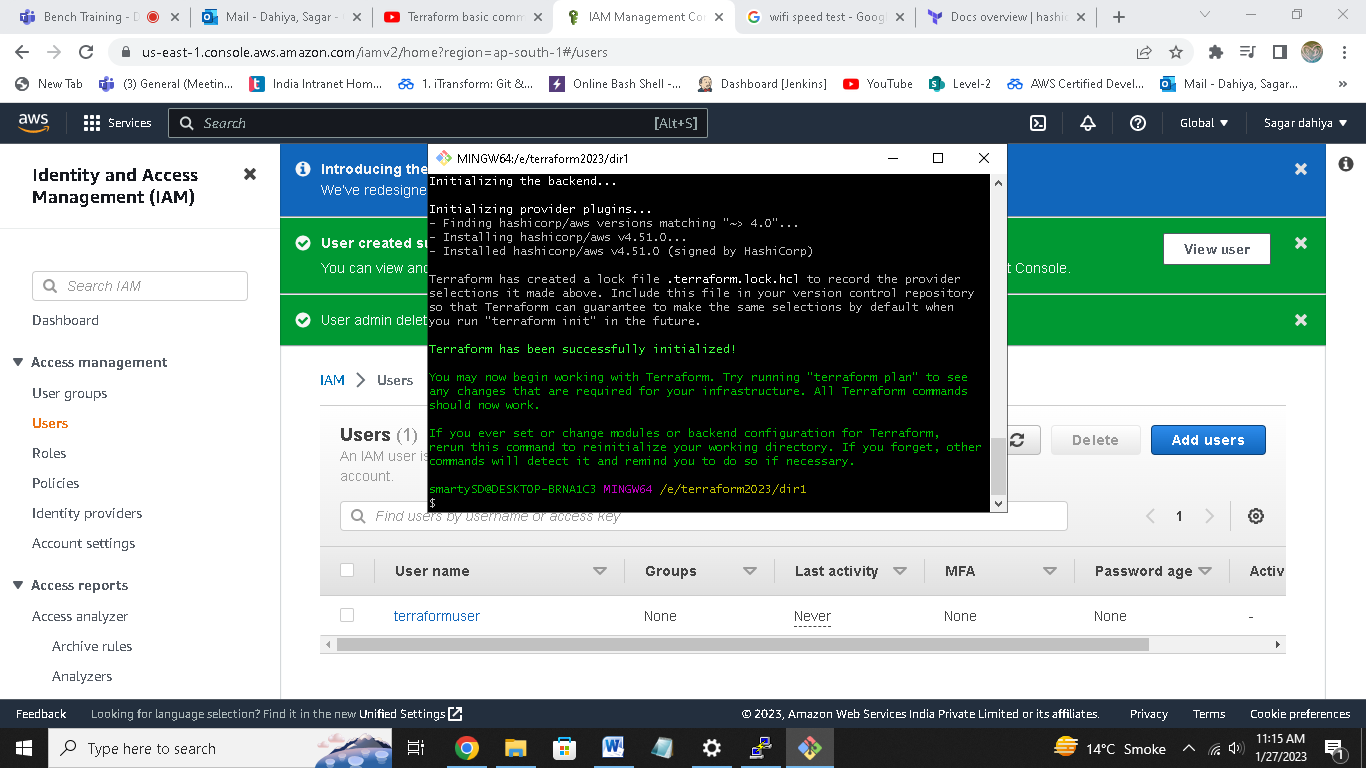
**version = "~> 4.0"**

**}**

**}**

**}**

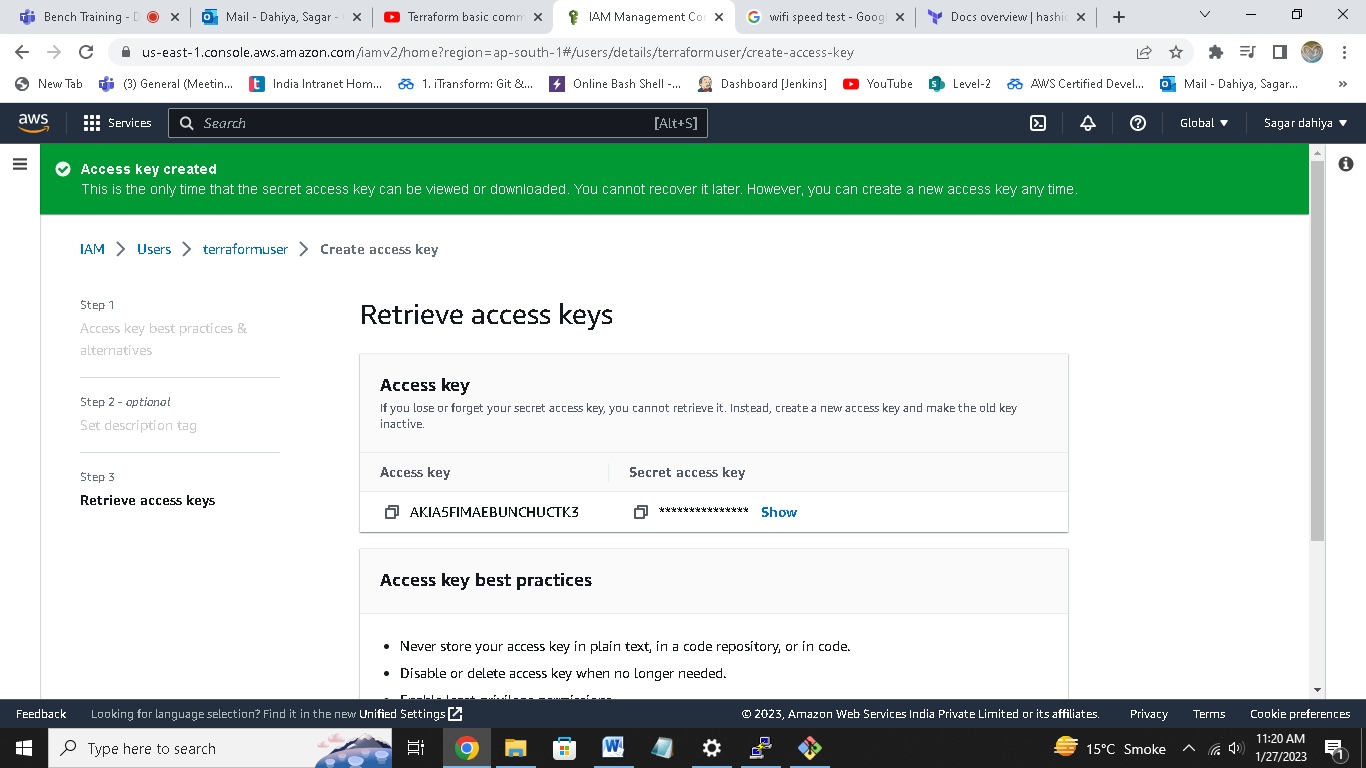
**Run “Terraform init”**

****

**CONNECT AWS TO TERRAFORM  
  
create new user in aws iam.**

**Give it administrator access**

**Save key pair value.**

****

**provider "aws" {**

**region = "us-west-2"**

**access\_key = "my-access-key"**

**secret\_key = "my-secret-key"**

**}**

**Store it in main.tf file**

**####to make new aws instance**

**resource "aws\_instance" "s1" {**

**ami = "ami-06984ea821ac0a879"**

**instance\_type = "t2.micro"**

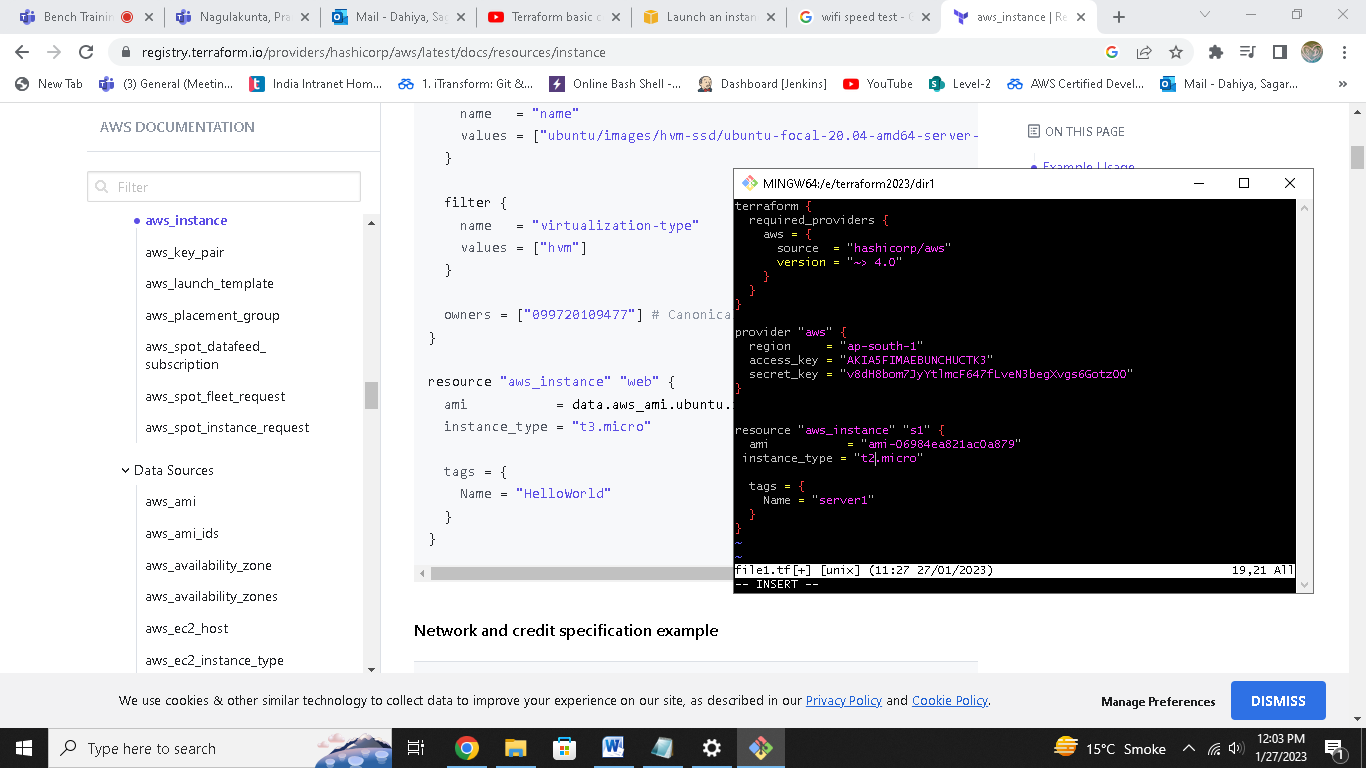
**tags = {**

**Name = "server1"**

**}**

**}**

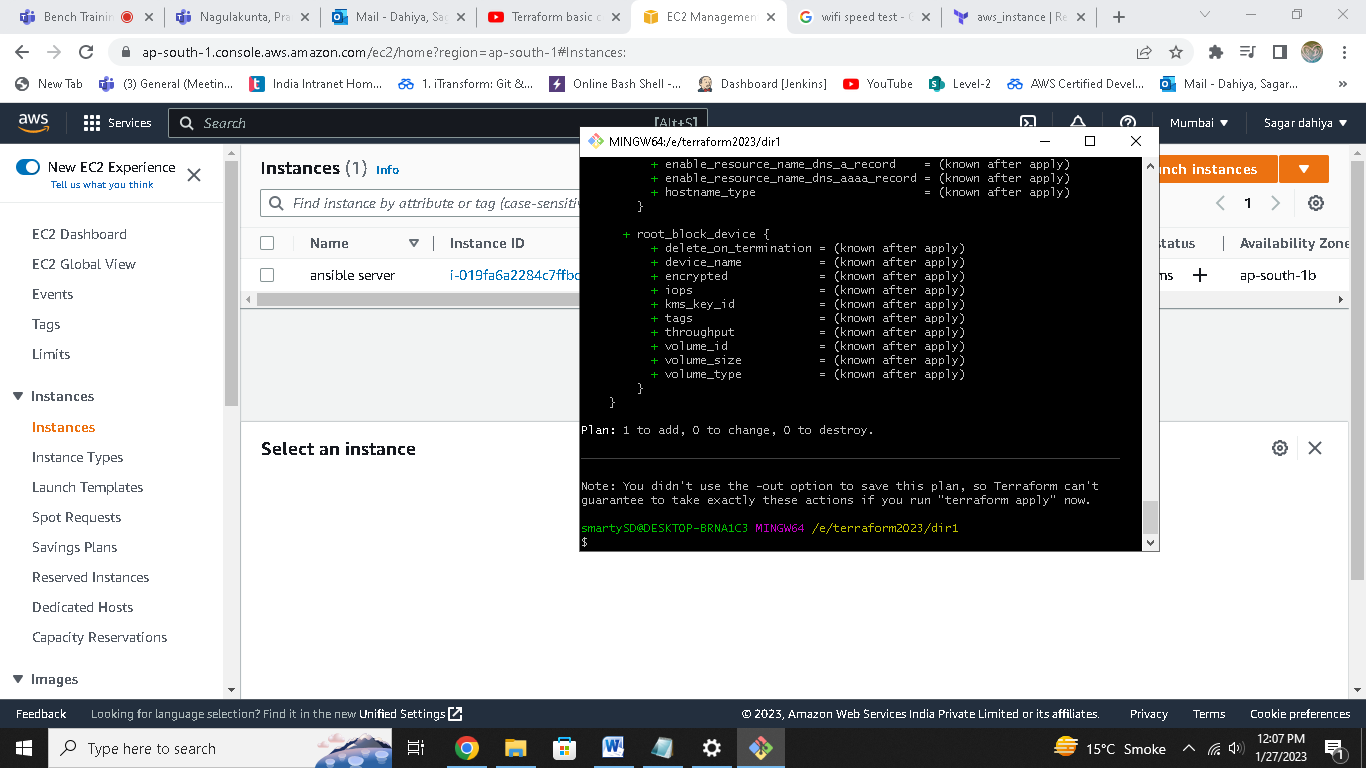
**Store it in main.tf**

****

**Terraform validate**

**# to check syntax is correct or not**

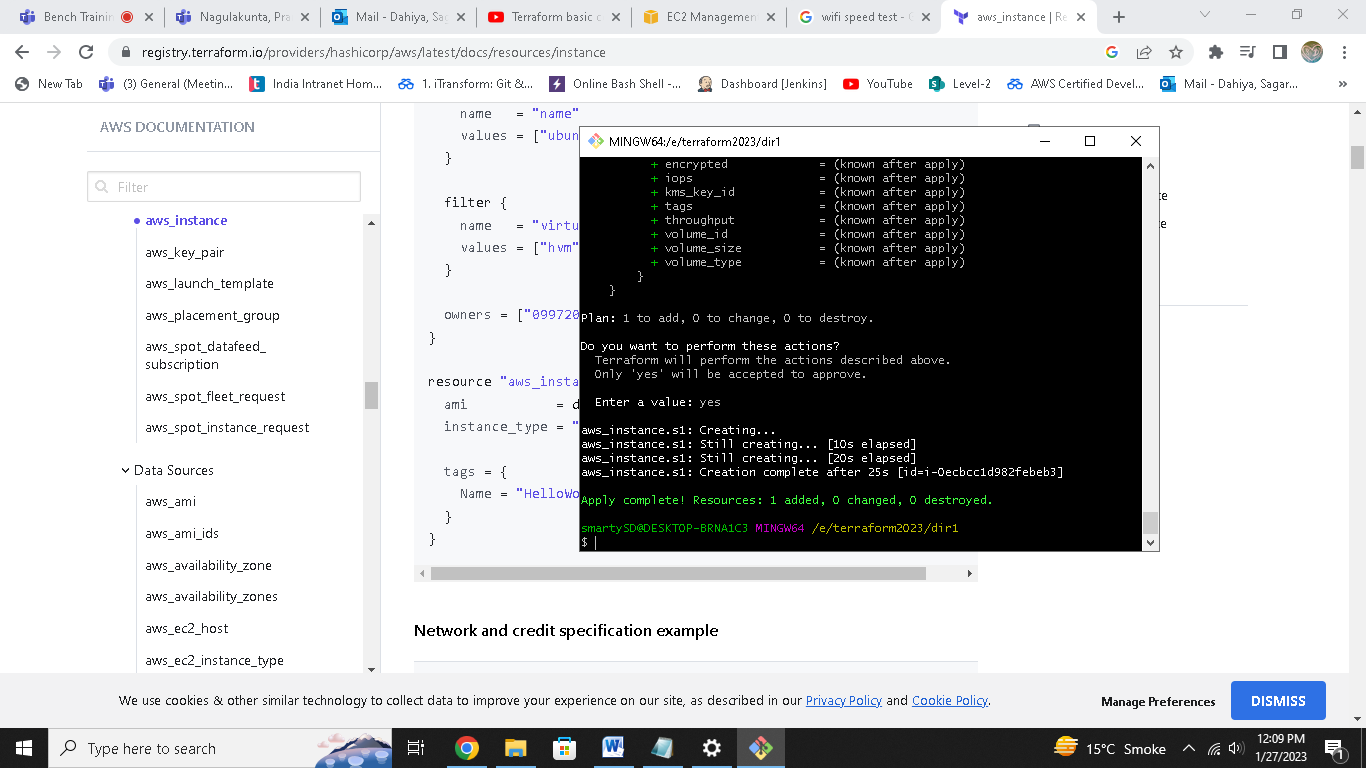
**Terraform plan**

****

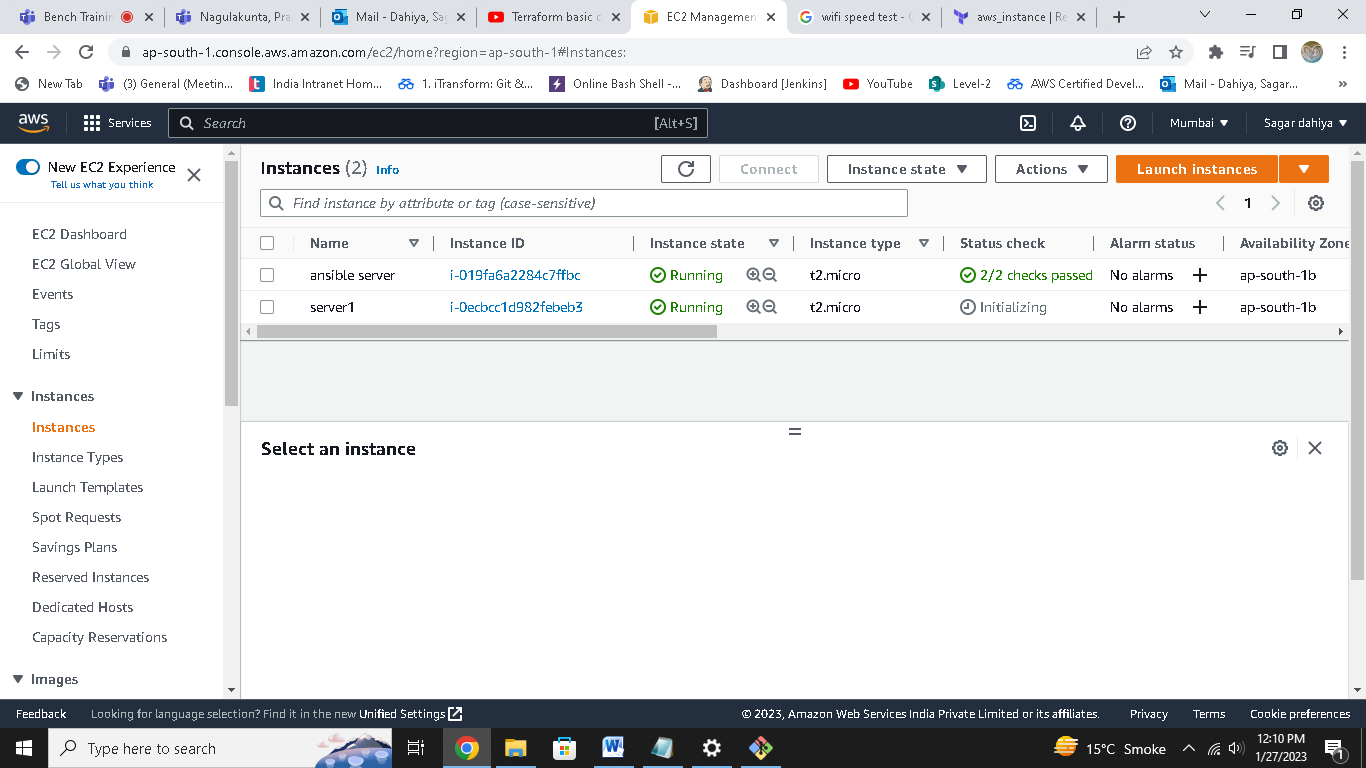
**Currently there is only 1 instance running.**

**Now run**

**Terraform apply**

****

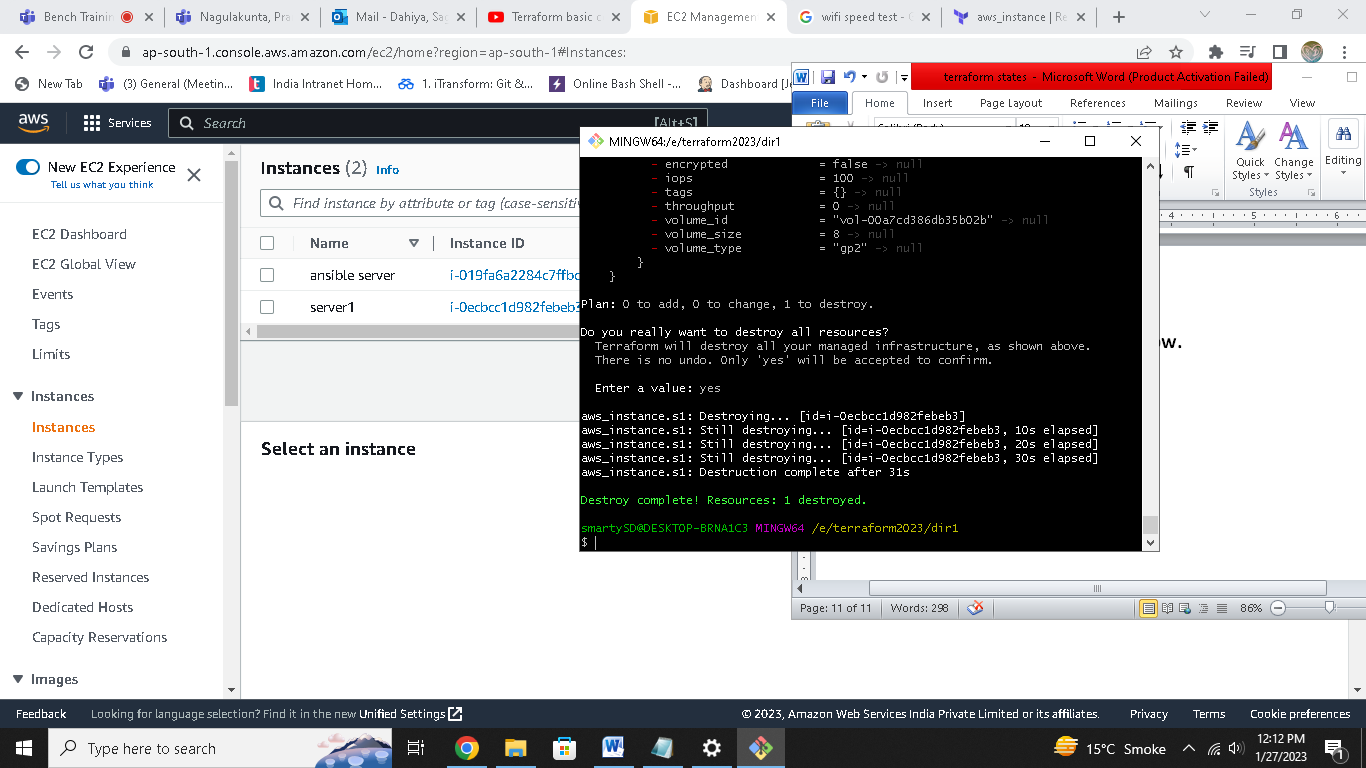
**Now check aws instances**

****

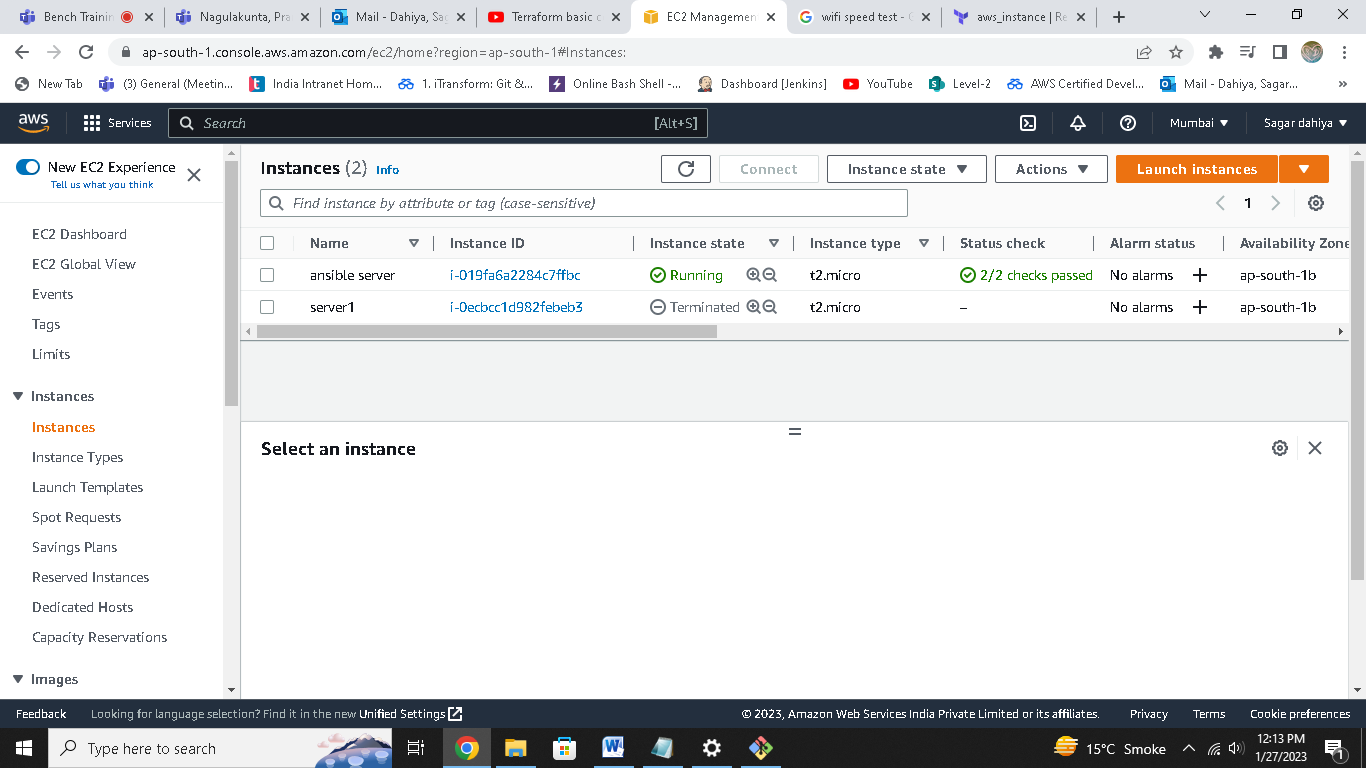
**You can see our server1 is running now.**

**Now to delete it,run**

**Terraform destroy**

****

**Our server destroyed**

****

**If you want to create 3 server at once**

resource "aws\_instance" "s1" {

ami = "ami-06984ea821ac0a879"

instance\_type = "t2.micro"

tags = {

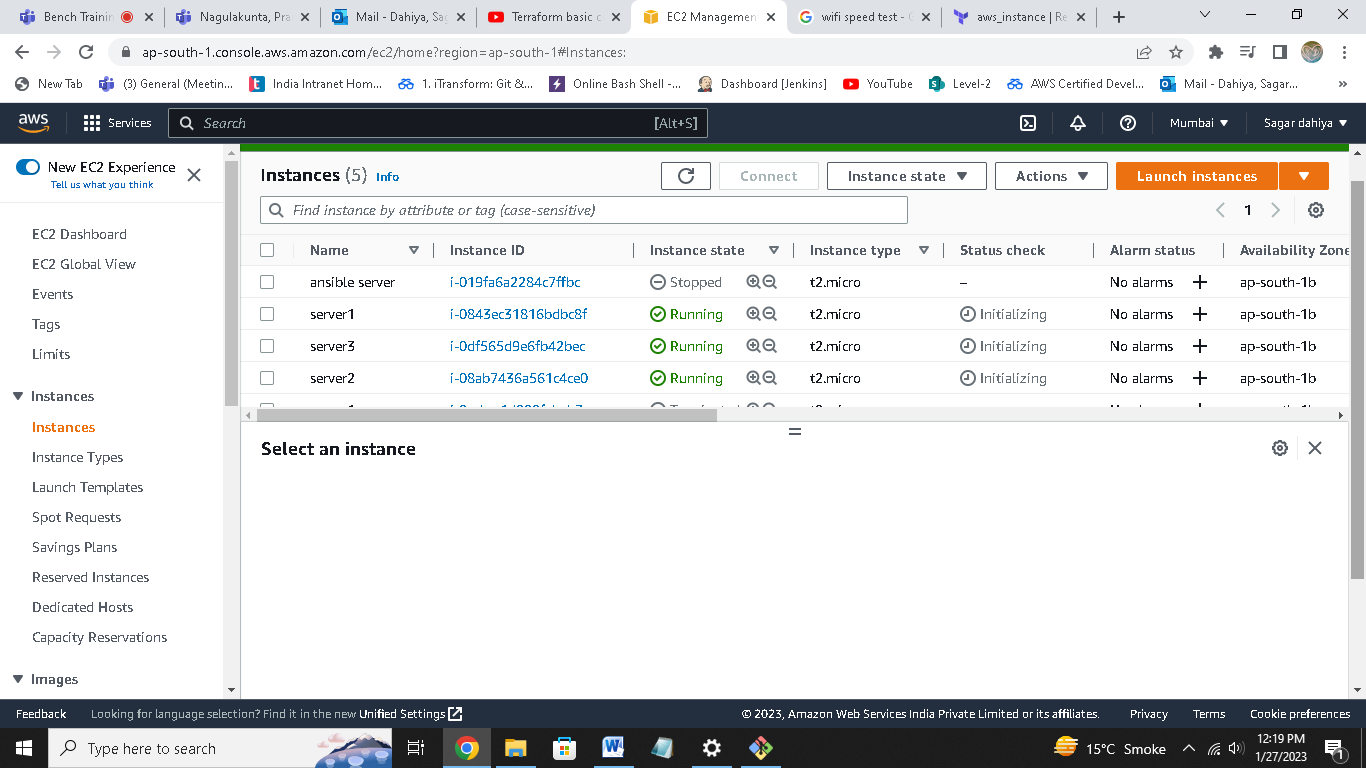
Name = "server1"

}

}

**Copy paste it 3 times**

**Now again terraform apply**

****

**3 servers are running.**