Create 2 EC2 instances on 2 different regions and install nginx using terraform script.

STEP 1 - write a terraform code for provider, user data, resource in a single main.tf file for launch 2 instance with nginx installation

```
# provider for sydney
provider "aws" {
    region = "ap-southeast-2" # Default regionterr
    alias = "ap-southeast-2"
}
# provider for singapore
provider "aws" {
    region = "ap-southeast-1" # Default regionterr
    alias = "ap-southeast-1" # Default regionterr
    alias = "ap-southeast-1"
```

```
# Launch EC2 Instance in us-east-2 with nginx
resource "aws instance" "singapore" {
  provider = aws.ap-southeast-1
                 = "ami-0f935a2ecd3a7bd5c"
  ami
  instance type = "t2.micro"
  user data = "#!/bin/bash
               sudo yum update S
               sudo yum install nfinx -y
               sudo service nginx status
               sudo service nginx start
               nginx --version"
  tags = {
   Name = "nginx"
output "public_ip" {
  value = aws instance.us-east-2.public ip
```

STEP 2 - terraform init & terraform validate for module and dependency and plugins verify

```
resource "aws instance" "us-east-1
PROBLEMS 8
             OUTPUT DEBUG CONSOLE
                                      TERMINAL
PS C:\test> terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.70.0
Terraform has been successfully initialized!
You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.
If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
PS C:\test> terraform validate
Success! The configuration is valid.
```

STEP 3 - terraform plan to execute a plan for creating instance as we mentioned in code

```
+ root_block_device (known after apply)
}

Plan: 2 to add, 0 to change, 0 to destroy.
```

STEP 4 - terraform apply to launch the instance using given datas

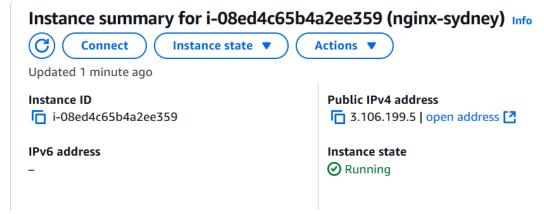
```
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_instance.us-east-1: Creating...
aws_instance.us-east-2: Creating...
aws_instance.us-east-1: Still creating... [10s elapsed]
aws_instance.us-east-2: Still creating... [10s elapsed]
aws_instance.us-east-2: Creation complete after 16s [id=i-09d11d3893ed8356d]
aws_instance.us-east-1: Creation complete after 17s [id=i-0f02e99d62136a21a]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
```

STEP - 5 - sydney{ap-southeast-2} running



STEP 6 - singapore{ap-southeast-1} running

Instance summary for i-0be2e802cc0f2c60b (nginx-singapore) Info

Updated less than a minute ago

Instance ID

i-0be2e802cc0f2c60b

IPv6 address

_

Public IPv4 address

13.215.51.35 | open address [2]

Instance state

Running

