

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

```
1  # provider for sydney
2  provider "aws" {
3      region = "ap-southeast-2" # Default region
4      alias = "ap-southeast-2"
5  }
6  # provider for singapore
7  provider "aws" {
8      region = "ap-southeast-1" # Default region
9      alias = "ap-southeast-1"
10 }
```

```
# Launch EC2 Instance in us-east-1 with nginx
resource "aws_instance" "sydney" {
    provider = aws.ap-southeast-2
    ami      = "ami-0146fc9ad419e2cfd"
    instance_type = "t2.micro"
    user_data = "#!/bin/bash
                sudo yum update -y
                sudo yum install nginx -y
                sudo service nginx status
                sudo service nginx start
                nginx --version"
    tags = {
        Name = "nginx"
    }
}
output "publicip" {
    value = aws_instance.us-east-1.public_ip
}
```

```
# Launch EC2 Instance in us-east-2 with nginx
resource "aws_instance" "singapore" {
  provider = aws.ap-southeast-1
  ami      = "ami-0f935a2ecd3a7bd5c"
  instance_type = "t2.micro"
  user_data = "#!/bin/bash
              sudo yum update -y
              sudo yum install nginx -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

```
2 resource "aws_instance" "us-east-1" {
```

PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS

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

### Instance summary for i-08ed4c65b4a2ee359 (nginx-sydney) [Info](#)


[Connect](#)[Instance state ▼](#)[Actions ▼](#)

Updated 1 minute ago

**Instance ID**  
 i-08ed4c65b4a2ee359

**Public IPv4 address**  
 3.106.199.5 | [open address](#) 

**IPv6 address**  
—

**Instance state**  
 **Running**

```

#_
~\#####_      Amazon Linux 2023
~~\#####\
~~\###|
~~\#/      https://aws.amazon.com/linux/amazon-linux-2023
~~V~' '~>
~~~
~~~.~.~
~~~\m/'
Last login: Thu Nov 28 05:46:26 2024 from 3.0.5.36
[ec2-user@ip-172-31-29-153 ~]$ nginx -v
nginx version: nginx/1.26.2
[ec2-user@ip-172-31-29-153 ~]$ █


```

## 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](#) 

### Instance state

Running

```
#_
~\   ###_           Amazon Linux 2023
~~ \-#####\
~~    \###|
~~      \#/         https://aws.amazon.com/linux/amazon-linux-2023
~~~~        v~-'->
~~~~
~~.-.-./
     /-/ -/-
       _/_m/'
```

Last login: Thu Nov 28 05:53:44 2024 from 3.0.5.36  
[ec2-user@ip-172-31-21-249 ~]\$ nginx -v  
nginx version: nginx/1.26.2  
[ec2-user@ip-172-31-21-249 ~]\$ █