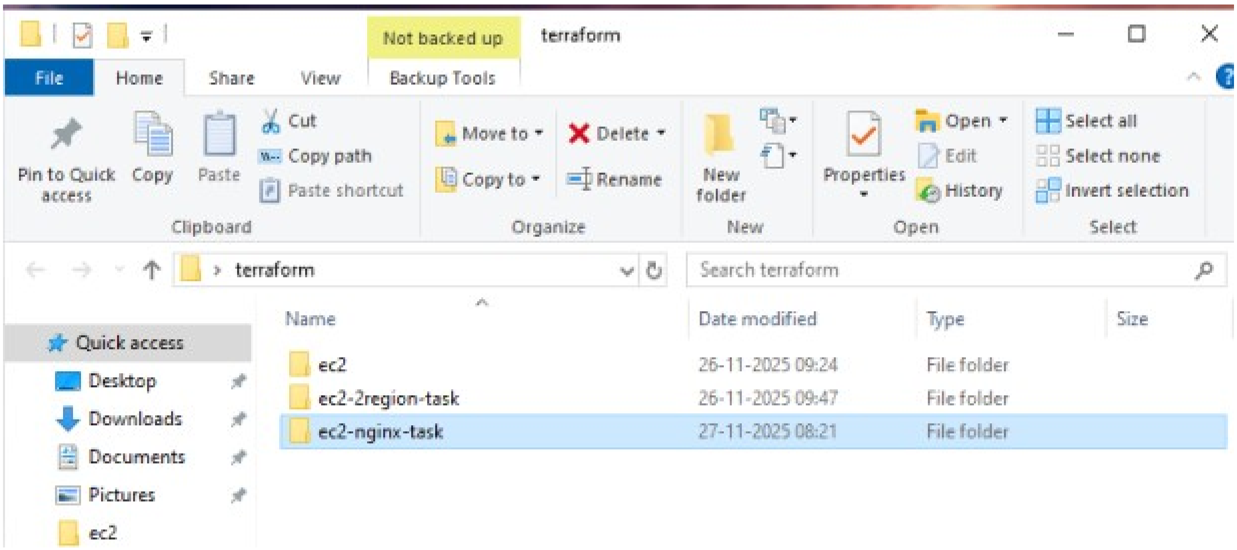
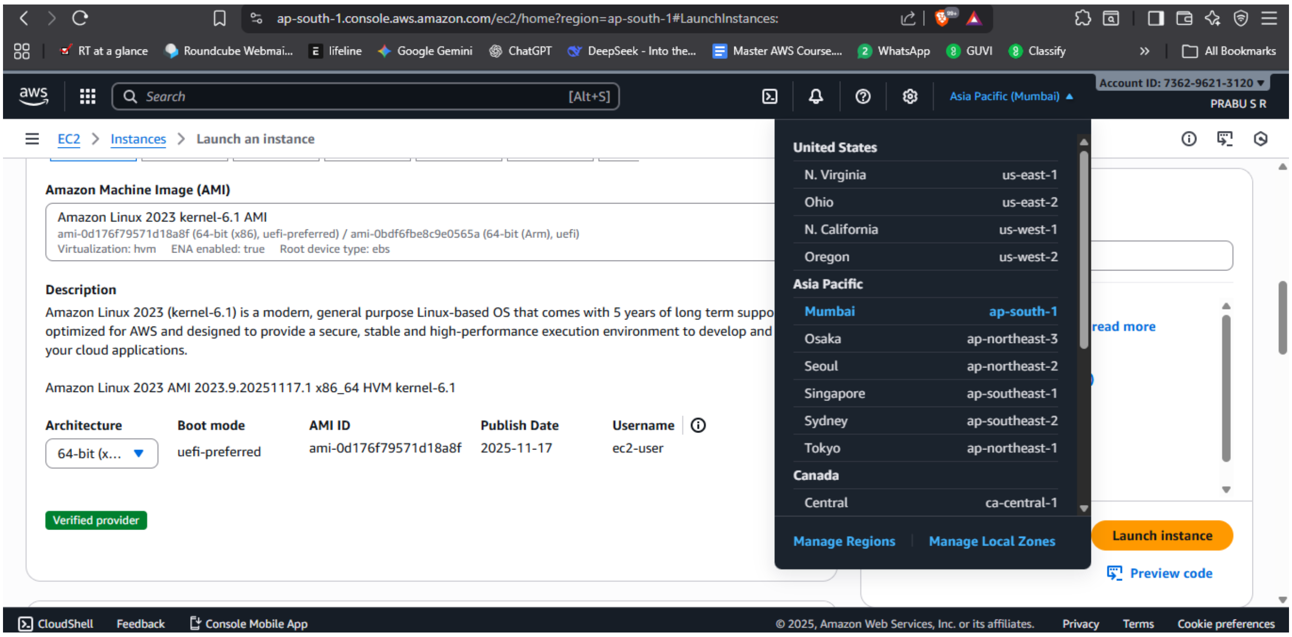
Terraform Task-2

Task Description:

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



Name	Date modified	Type	Size
ec2	26-11-2025 09:24	File folder	
ec2-2region-task	26-11-2025 09:47	File folder	
ec2-nginx-task	27-11-2025 08:21	File folder	



Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI
ami-0d176f79571d18a8f (64-bit (x86), uefi-preferred) / ami-0bdf6fbe8c9e0565a (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

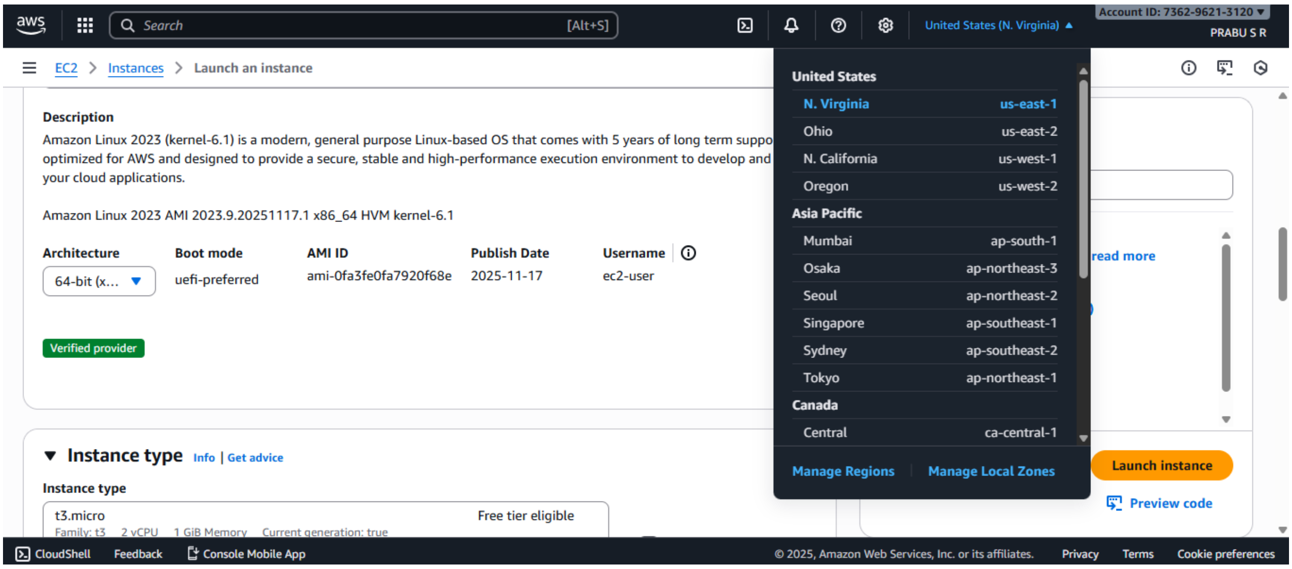
Description

Amazon Linux 2023 (kernel-6.1) is a modern, general purpose Linux-based OS that comes with 5 years of long term suppo optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and your cloud applications.

Amazon Linux 2023 AMI 2023.9.20251117.1 x86_64 HVM kernel-6.1

Architecture	Boot mode	AMI ID	Publish Date	Username
64-bit (x...	uefi-preferred	ami-0d176f79571d18a8f	2025-11-17	ec2-user

United States	
N. Virginia	us-east-1
Ohio	us-east-2
N. California	us-west-1
Oregon	us-west-2
Asia Pacific	
Mumbai	ap-south-1
Osaka	ap-northeast-3
Seoul	ap-northeast-2
Singapore	ap-southeast-1
Sydney	ap-southeast-2
Tokyo	ap-northeast-1
Canada	
Central	ca-central-1



aws
Search [Alt+S]
United States (N. Virginia) ▲
Account ID: 7362-9621-3120 ▾
PRABU S R
EC2 > Instances > Launch an instance

Description
Amazon Linux 2023 (kernel-6.1) is a modern, general purpose Linux-based OS that comes with 5 years of long term suppo
optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and
your cloud applications.

Amazon Linux 2023 AMI 2023.9.20251117.1 x86_64 HVM kernel-6.1

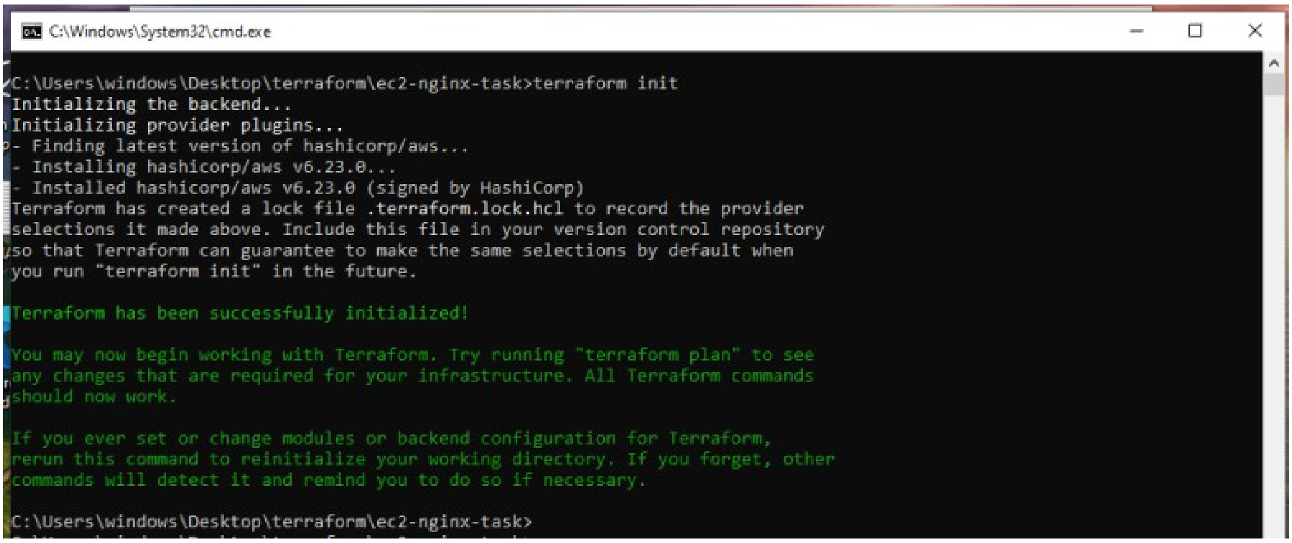
Architecture	Boot mode	AMI ID	Publish Date	Username
64-bit (x... ▾	uefi-preferred	ami-0fa3fe0fa7920f68e	2025-11-17	ec2-user

Verified provider

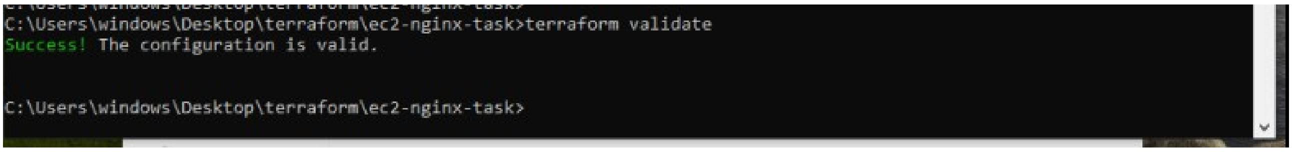
Instance type Info | Get advice

Instance type
t3.micro Free tier eligible
Family: t3 2 vCPU 1 GiB Memory Current generation: true

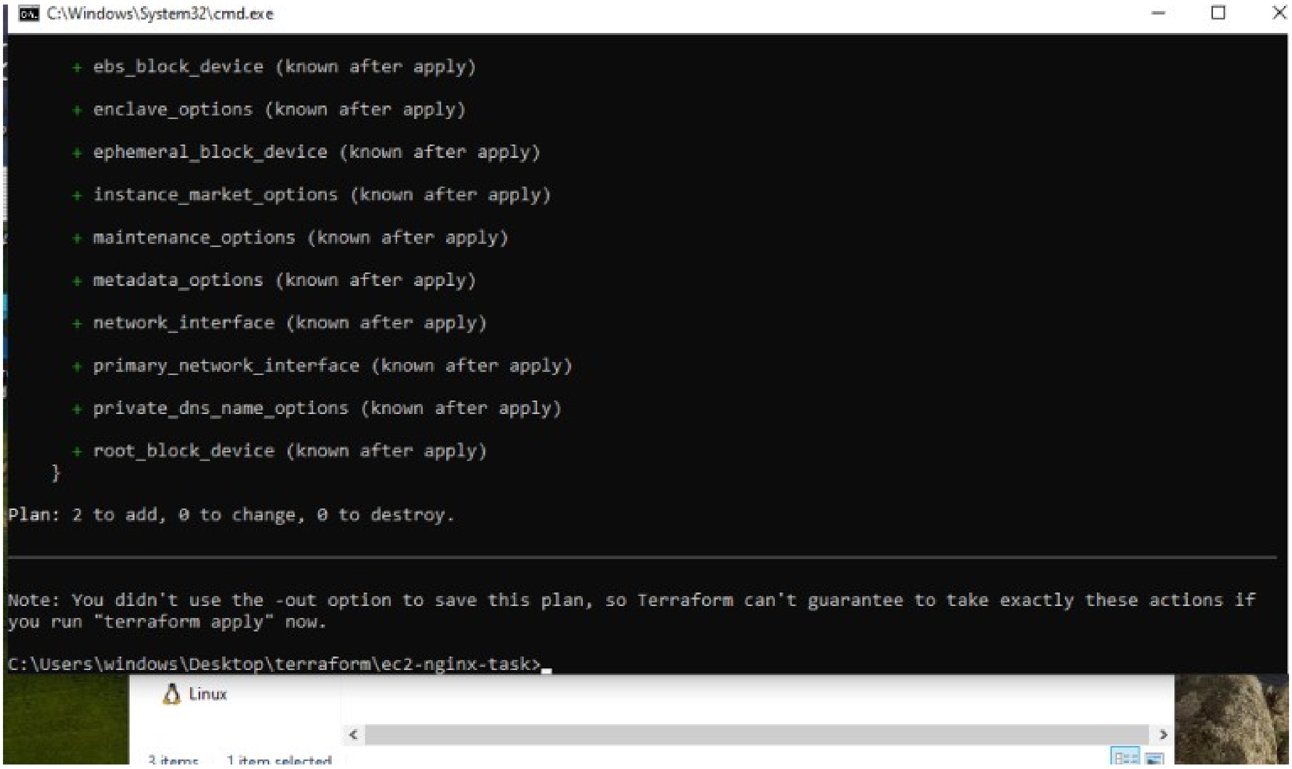
United States	
N. Virginia	us-east-1
Ohio	us-east-2
N. California	us-west-1
Oregon	us-west-2
Asia Pacific	
Mumbai	ap-south-1
Osaka	ap-northeast-3
Seoul	ap-northeast-2
Singapore	ap-southeast-1
Sydney	ap-southeast-2
Tokyo	ap-northeast-1
Canada	
Central	ca-central-1



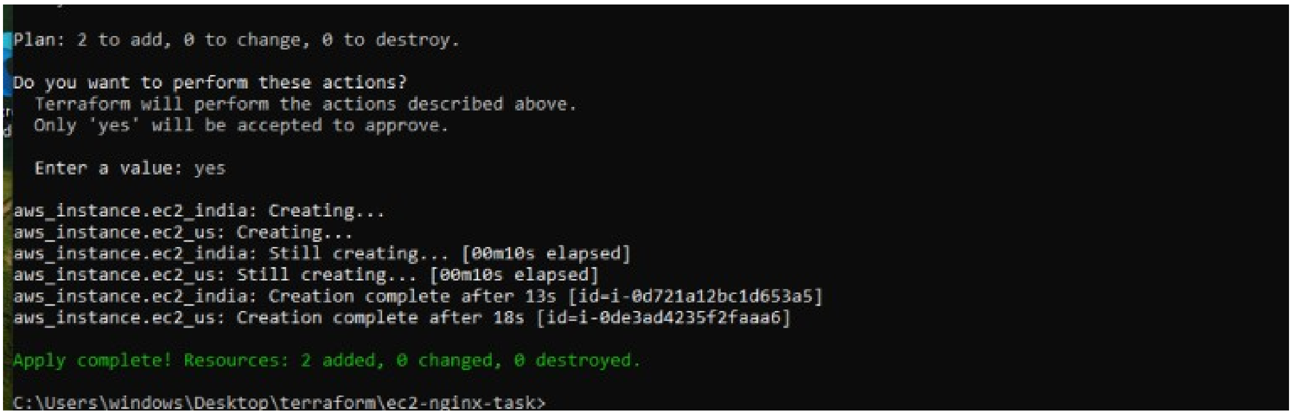
```
C:\Users\windows\Desktop\terraform\ec2-nginx-task>terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v6.23.0...
- Installed hashicorp/aws v6.23.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

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.

C:\Users\windows\Desktop\terraform\ec2-nginx-task>
```



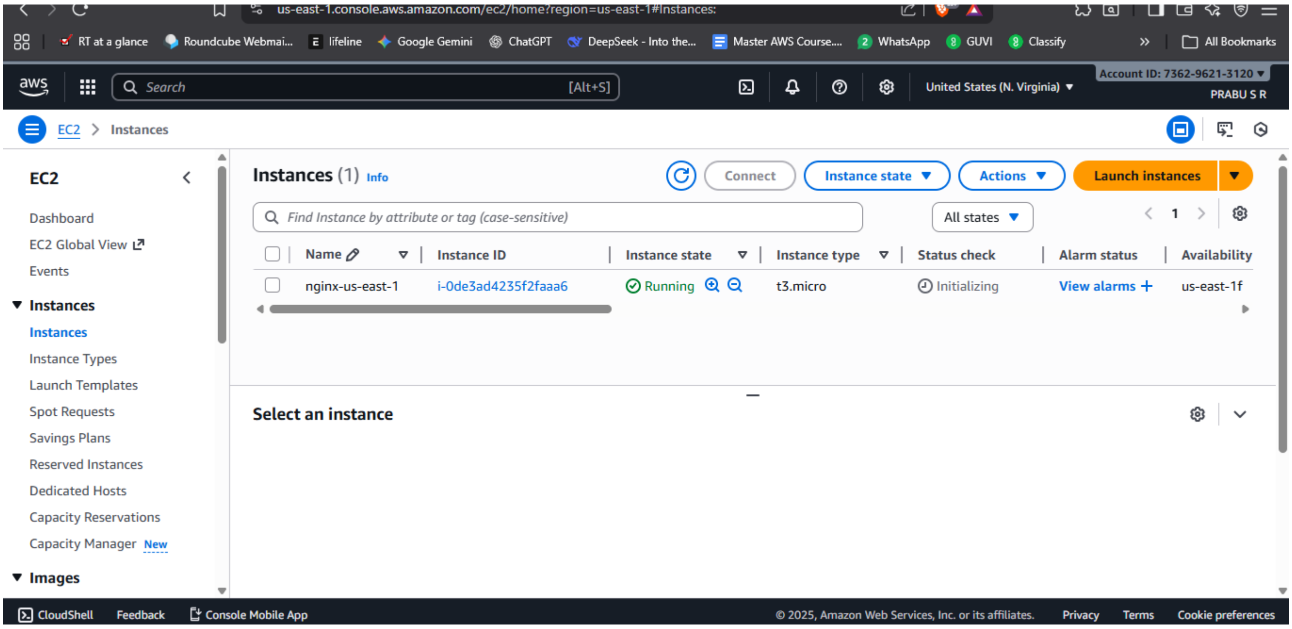
```
C:\Users\windows\Desktop\terraform\ec2-nginx-task>terraform validate
Success! The configuration is valid.


C:\Users\windows\Desktop\terraform\ec2-nginx-task>
```



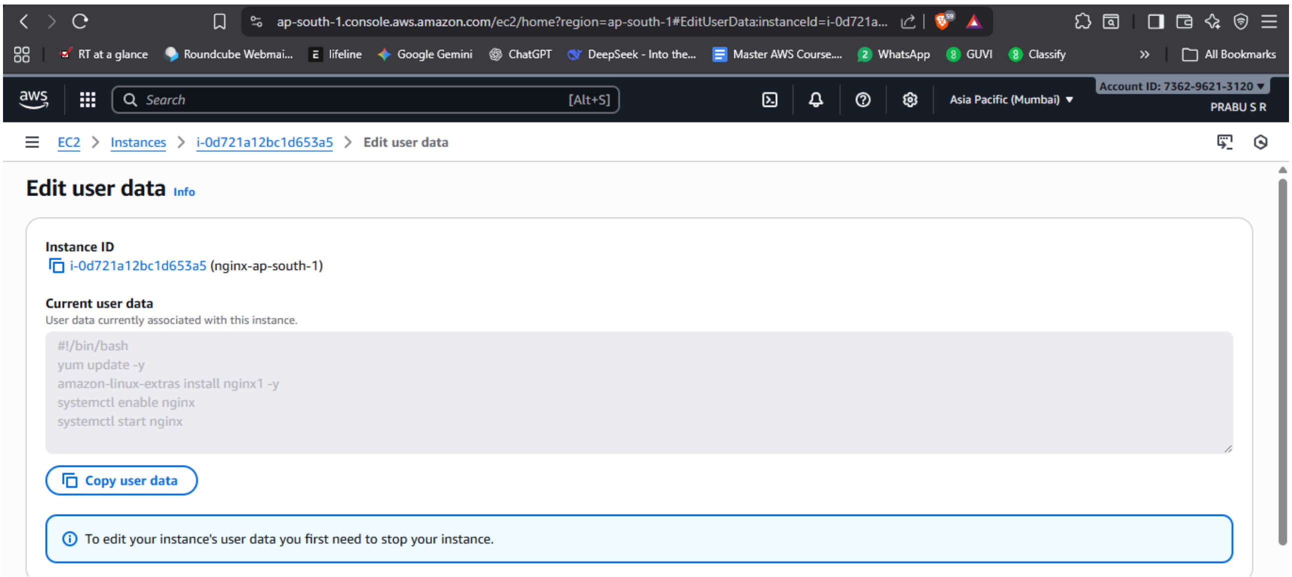
```
      + ebs_block_device (known after apply)

      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + primary_network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

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

─────────────────────────────────────────────────────────────────────────────────────────────

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if
you run "terraform apply" now.

C:\Users\windows\Desktop\terraform\ec2-nginx-task>_
```



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

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_instance.ec2_india: Creating...
aws_instance.ec2_us: Creating...
aws_instance.ec2_india: Still creating... [00m10s elapsed]
aws_instance.ec2_us: Still creating... [00m10s elapsed]
aws_instance.ec2_india: Creation complete after 13s [id=i-0d721a12bc1d653a5]
aws_instance.ec2_us: Creation complete after 18s [id=i-0de3ad4235f2faaa6]

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

C:\Users\windows\Desktop\terraform\ec2-nginx-task>
```



□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
□	nginx-us-east-1	i-0de3ad4235f2faaa6	✓ Running	t3.micro	Initializing	View alarms +	us-east-1f



Edit user data Info

Instance ID

i-0d721a12bc1d653a5 (nginx-ap-south-1)

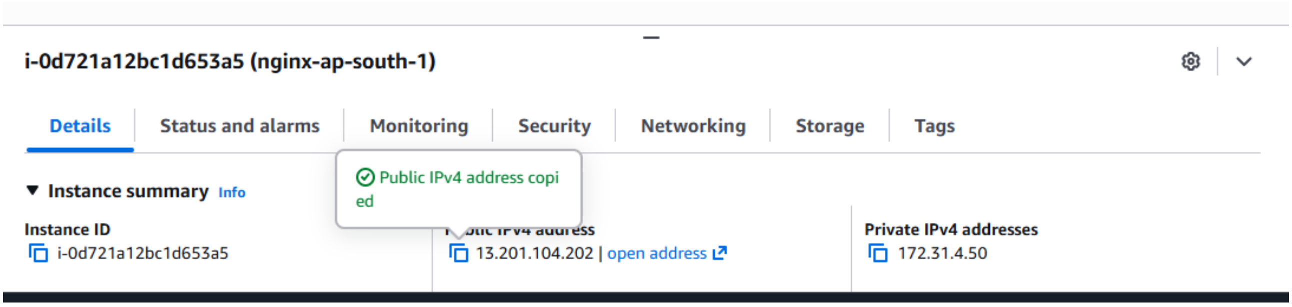
Current user data

User data currently associated with this instance.

```
#!/bin/bash
yum update -y
amazon-linux-extras install nginx1 -y
systemctl enable nginx
systemctl start nginx
```

Copy user data

To edit your instance's user data you first need to stop your instance.

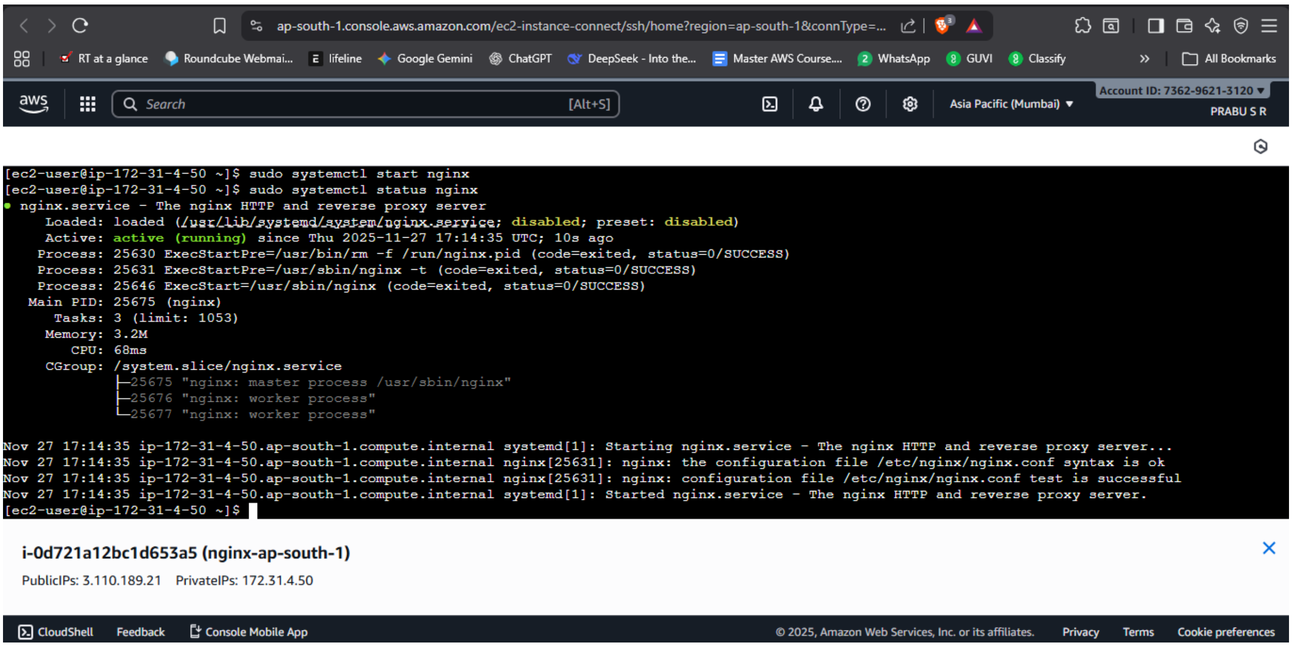


i-0d721a12bc1d653a5 (nginx-ap-south-1)

Instance ID: i-0d721a12bc1d653a5

Public IPv4 address: 13.201.104.202 | open address

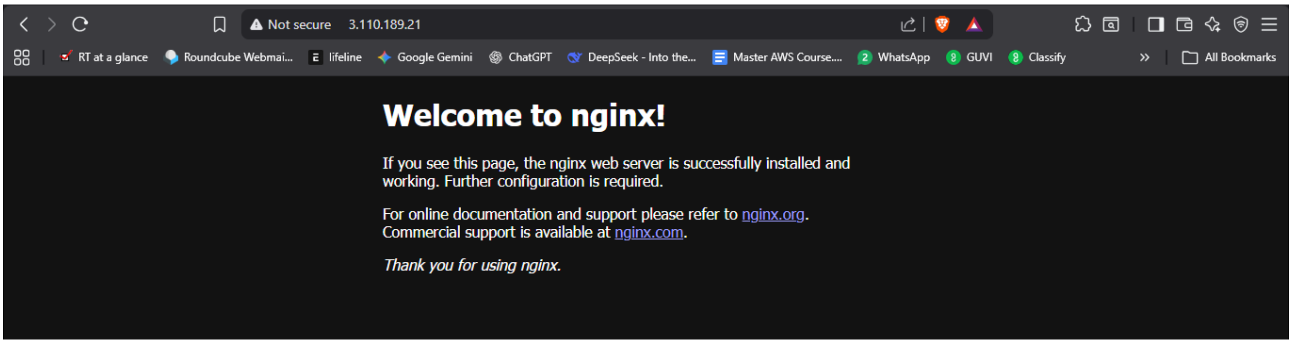
Private IPv4 addresses: 172.31.4.50



```
[ec2-user@ip-172-31-4-50 ~]$ sudo systemctl start nginx
[ec2-user@ip-172-31-4-50 ~]$ sudo systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; disabled; preset: disabled)
     Active: active (running) since Thu 2025-11-27 17:14:35 UTC; 10s ago
    Process: 25630 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
    Process: 25631 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
    Process: 25646 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
   Main PID: 25675 (nginx)
      Tasks: 3 (limit: 1053)
     Memory: 3.2M
        CPU: 68ms
     CGroup: /system.slice/nginx.service
             ├─25675 "nginx: master process /usr/sbin/nginx"
             ├─25676 "nginx: worker process"
             └─25677 "nginx: worker process"

Nov 27 17:14:35 ip-172-31-4-50.ap-south-1.compute.internal systemd[1]: Starting nginx.service - The nginx HTTP and reverse proxy server...
Nov 27 17:14:35 ip-172-31-4-50.ap-south-1.compute.internal nginx[25631]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Nov 27 17:14:35 ip-172-31-4-50.ap-south-1.compute.internal nginx[25631]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Nov 27 17:14:35 ip-172-31-4-50.ap-south-1.compute.internal systemd[1]: Started nginx.service - The nginx HTTP and reverse proxy server.
[ec2-user@ip-172-31-4-50 ~]$
```

i-0d721a12bc1d653a5 (nginx-ap-south-1)

PublicIPs: 3.110.189.21 PrivateIPs: 172.31.4.50



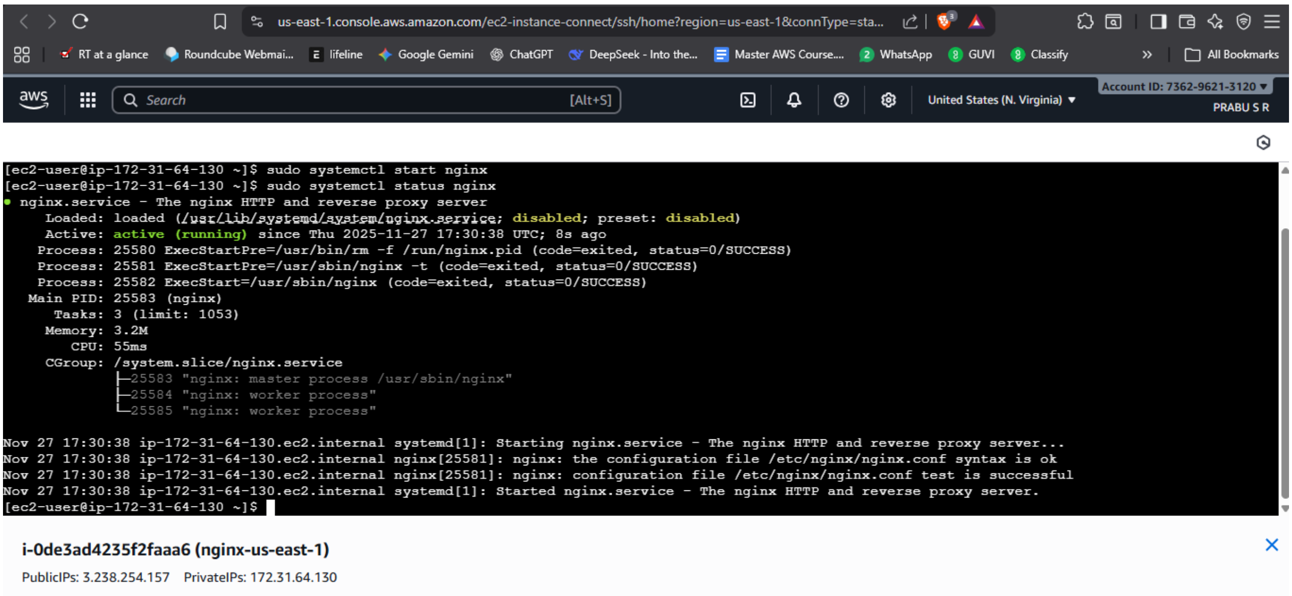
Not secure 3.110.189.21

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

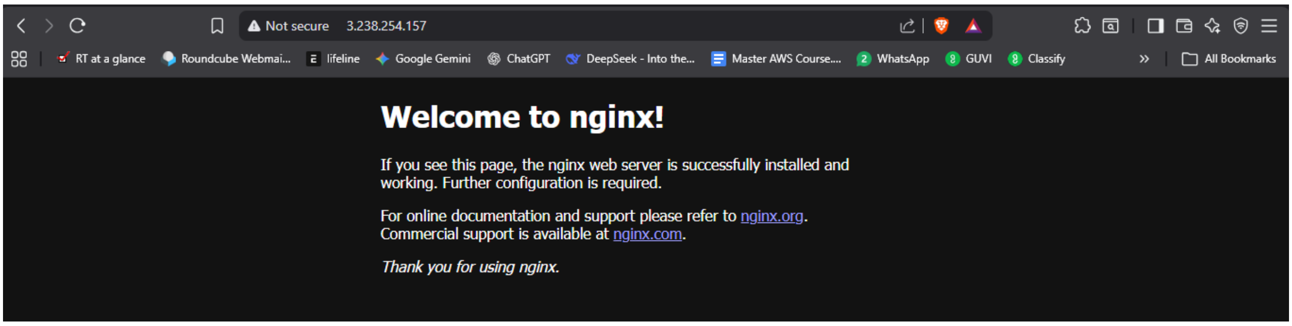


us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=us-east-1&connType=sta...

```
[ec2-user@ip-172-31-64-130 ~]$ sudo systemctl start nginx
[ec2-user@ip-172-31-64-130 ~]$ sudo systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; disabled; preset: disabled)
     Active: active (running) since Thu 2025-11-27 17:30:38 UTC; 8s ago
    Process: 25580 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
    Process: 25581 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
    Process: 25582 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
   Main PID: 25583 (nginx)
      Tasks: 3 (limit: 1053)
     Memory: 3.2M
        CPU: 55ms
     CGroup: /system.slice/nginx.service
             ├─25583 "nginx: master process /usr/sbin/nginx"
             ├─25584 "nginx: worker process"
             └─25585 "nginx: worker process"

Nov 27 17:30:38 ip-172-31-64-130.ec2.internal systemd[1]: Starting nginx.service - The nginx HTTP and reverse proxy server...
Nov 27 17:30:38 ip-172-31-64-130.ec2.internal nginx[25581]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Nov 27 17:30:38 ip-172-31-64-130.ec2.internal nginx[25581]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Nov 27 17:30:38 ip-172-31-64-130.ec2.internal systemd[1]: Started nginx.service - The nginx HTTP and reverse proxy server.
[ec2-user@ip-172-31-64-130 ~]$
```

i-0de3ad4235f2faaa6 (nginx-us-east-1)

PublicIPs: 3.238.254.157 PrivateIPs: 172.31.64.130



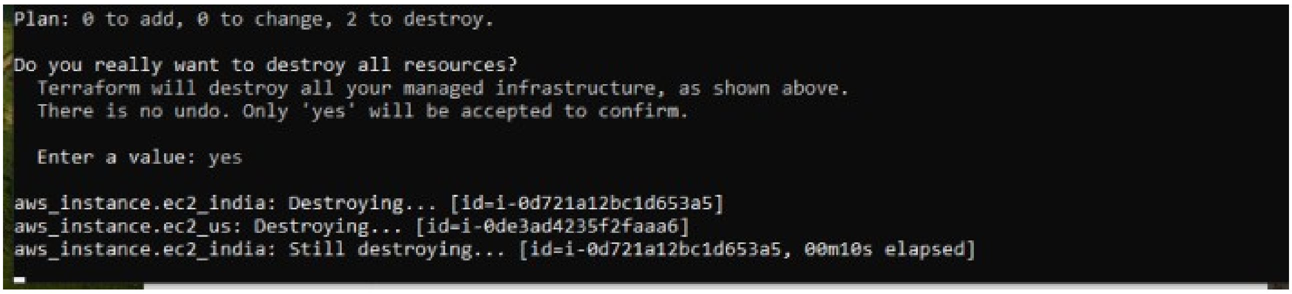
Not secure 3.238.254.157

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.



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

Do you really want to destroy all resources?
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.

  Enter a value: yes

aws_instance.ec2_india: Destroying... [id=i-0d721a12bc1d653a5]
aws_instance.ec2_us: Destroying... [id=i-0de3ad4235f2faaa6]
aws_instance.ec2_india: Still destroying... [id=i-0d721a12bc1d653a5, 00m10s elapsed]
```