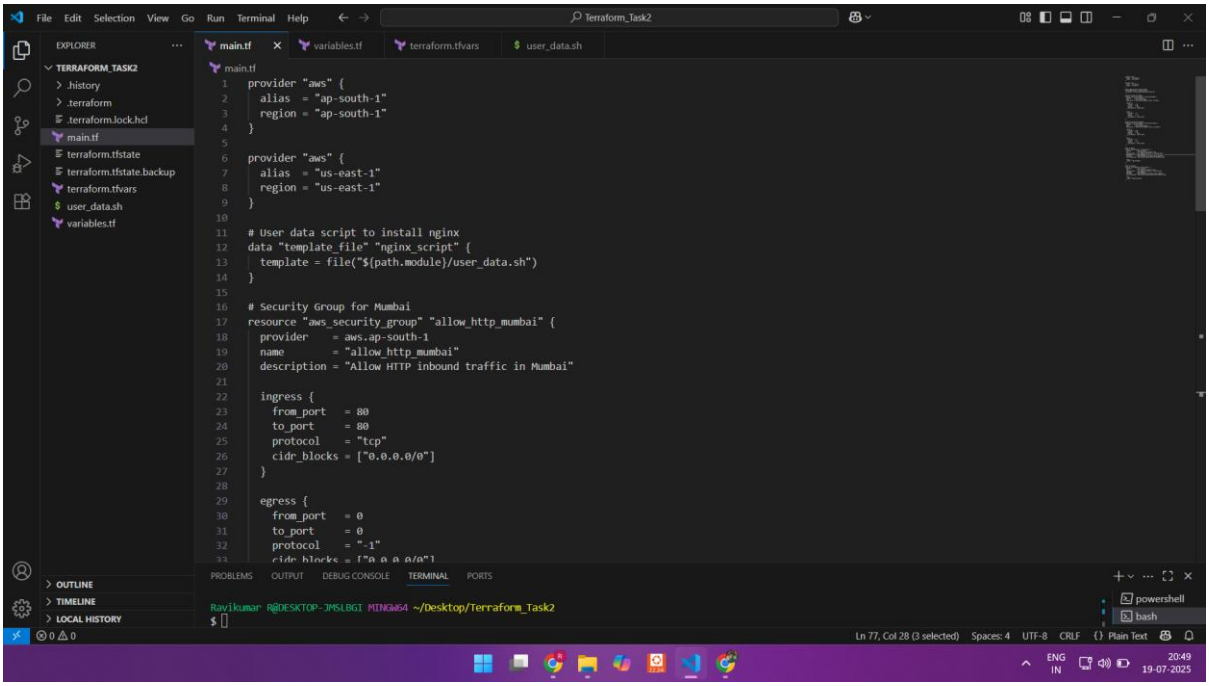
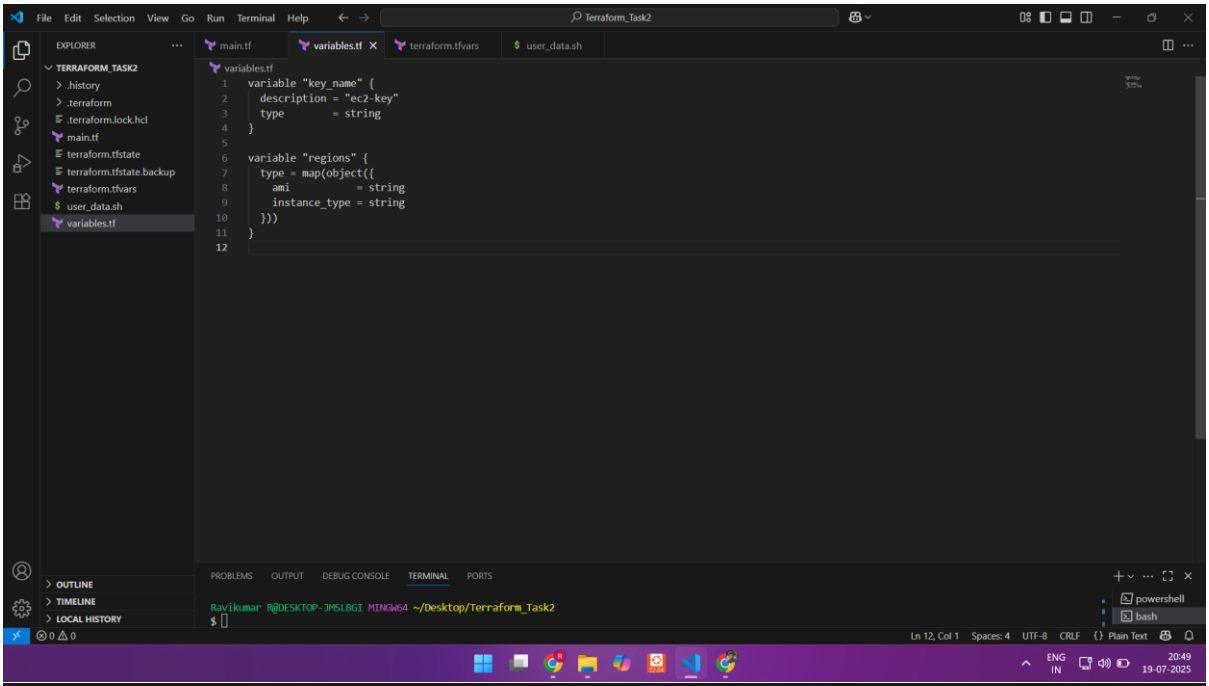
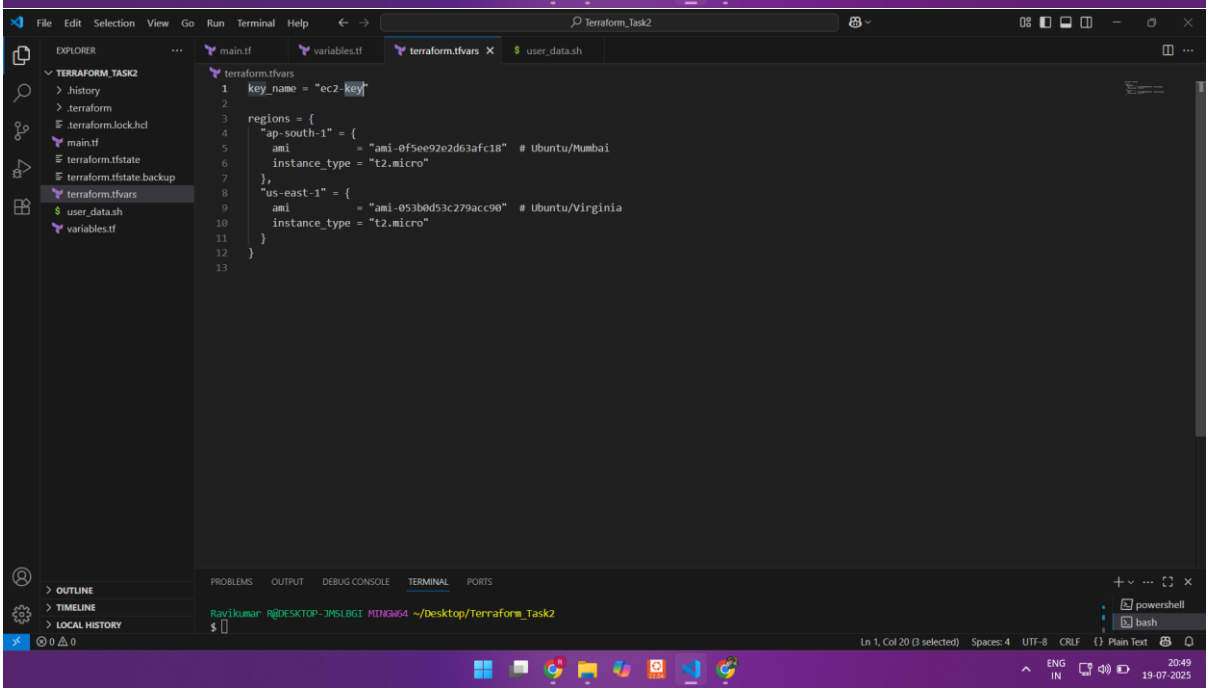
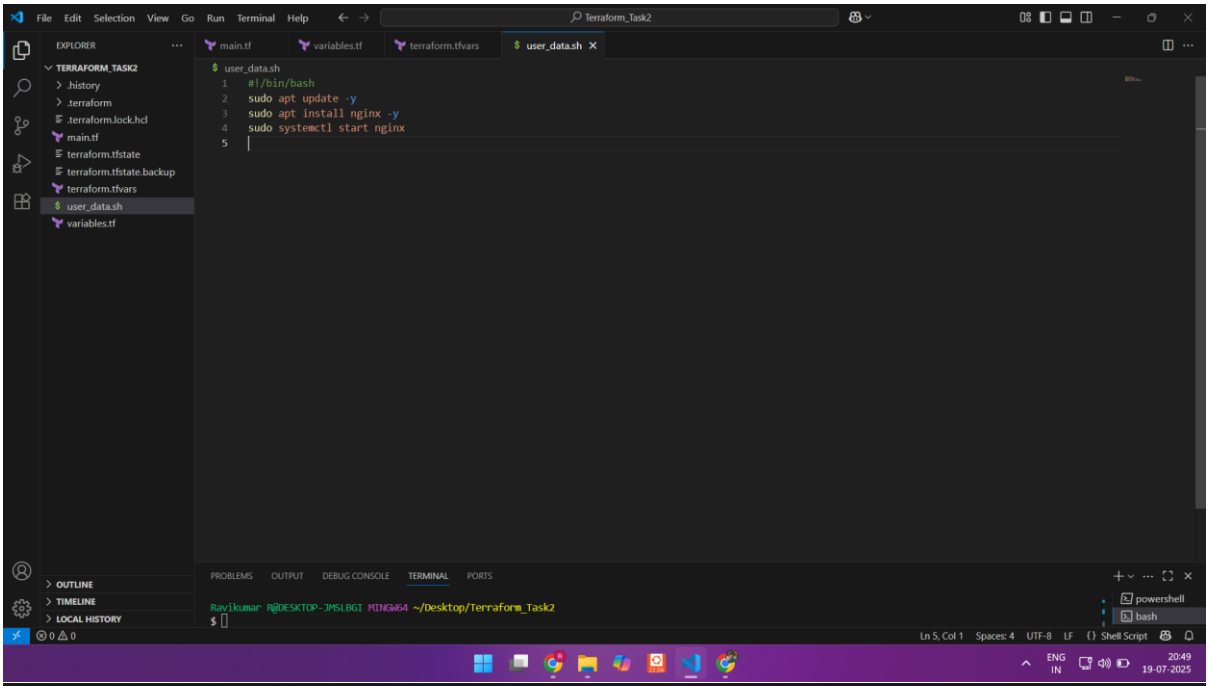
Create 2 EC2 instances on 2 different regions and install nginx using terraform script.

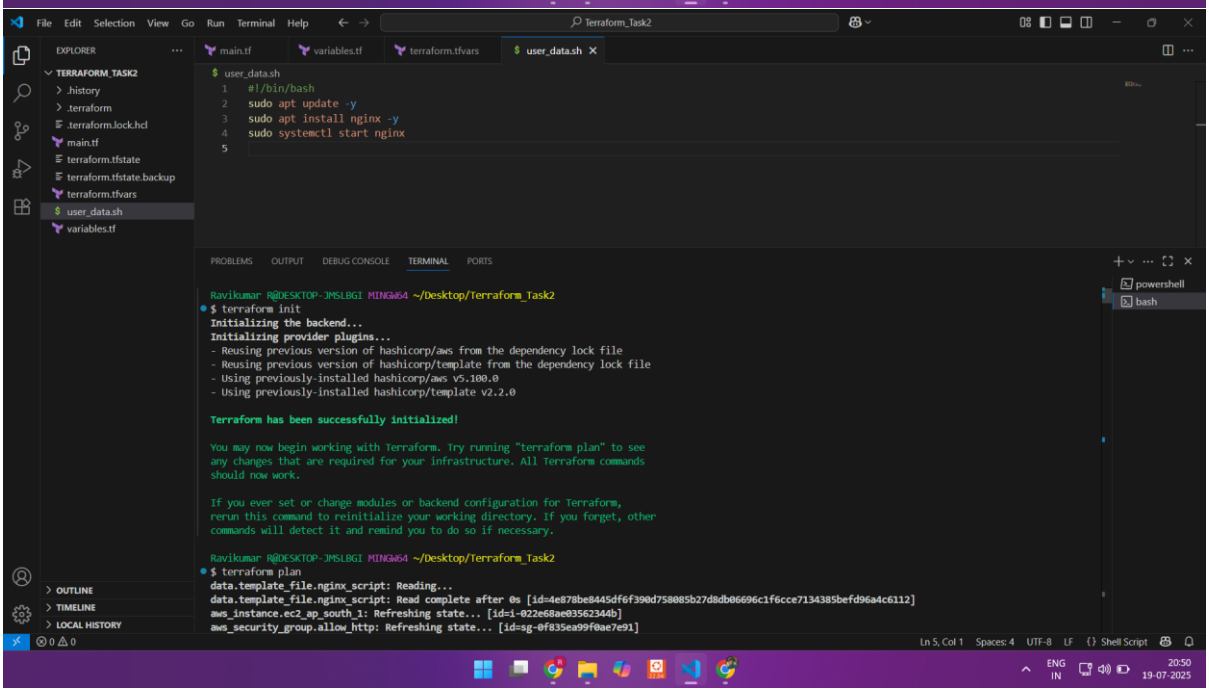
1. I Have created folder within that 4 files to be created main.tf, variables.tf, terraform.tfvars, user_data.sh. In main.tf I have written code to crete 2 ec2 instance with regions instance types etc.., and created to open the port 80 to 80 to enable http inbound rules from terraform for both regions,
2. Variables.sh file to declared the typrs of variable like strings, then in terraform.tfvars which regions, which types of instance, which ami I have declared and also key pair declared.
3. In user_data.sh I have written bash script nginx installation commands to install nginx. So in main.tf Both the regions calls to install the nginx I have declared.
4. Then in terminal first initialization (Terraform init), then Terraform plan to show what will be created, then Terraform apply to create instance.
5. After both the EC2 instance are installed, then checked in aws console and also checked the public ips for nginx installation.

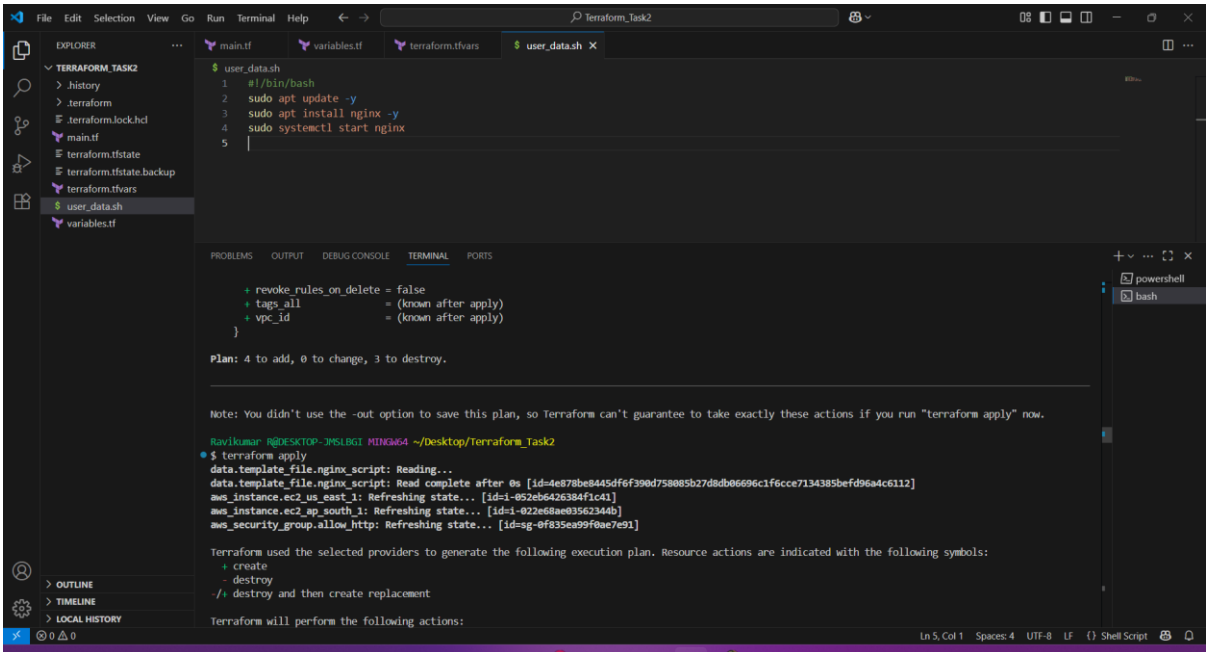


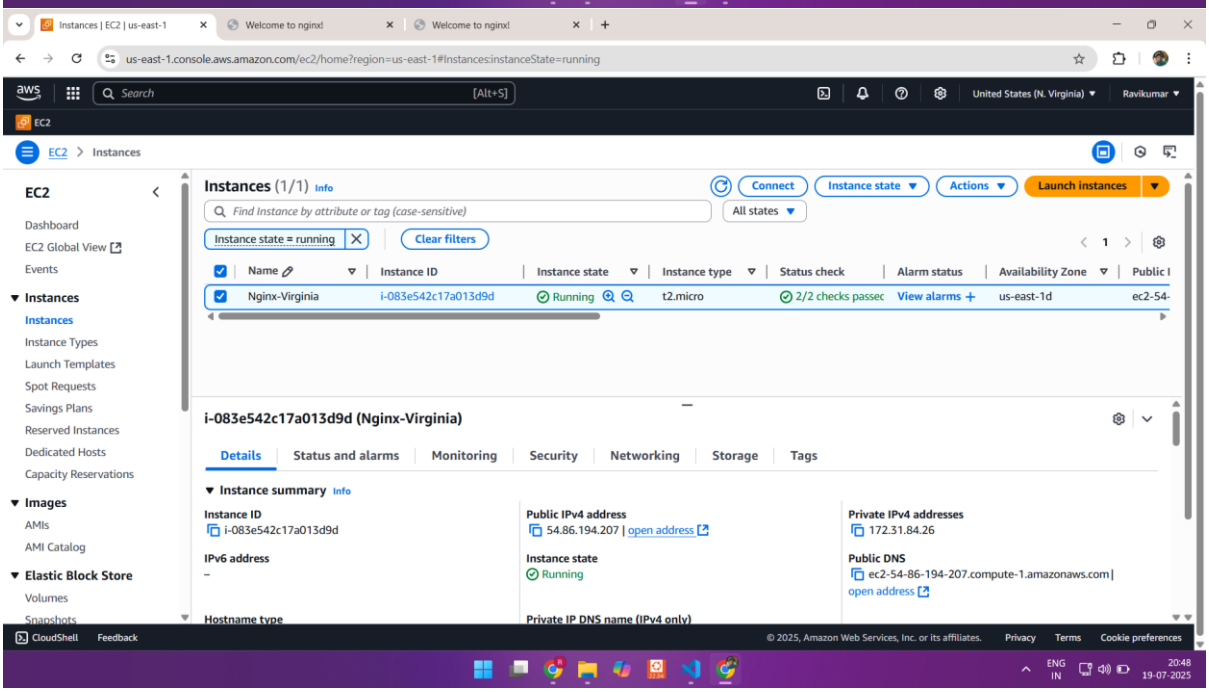

File Edit Selection View Go Run Terminal Help
Terraform_Task2
EXPLORER
TERRAFORM_TASK2
.history
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
terraform.tfvars
user_data.sh
variables.tf
main.tf
variables.tf
terraform.tfvars
user_data.sh
variables.tf
variable "key_name" {
description = "ec2-key"
type = string
}
variable "regions" {
type = map(object({
ami = string
instance_type = string
}))
}
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Ravikumar R@DESKTOP-JMSLBGI MINGW64 ~/Desktop/Terraform_Task2
$
powershell
bash
OUTLINE
TIMELINE
LOCAL HISTORY
Ln 12, Col 1 Spaces: 4 UTF-8 CRLF Plain Text
ENG IN
20:49
19-07-2025

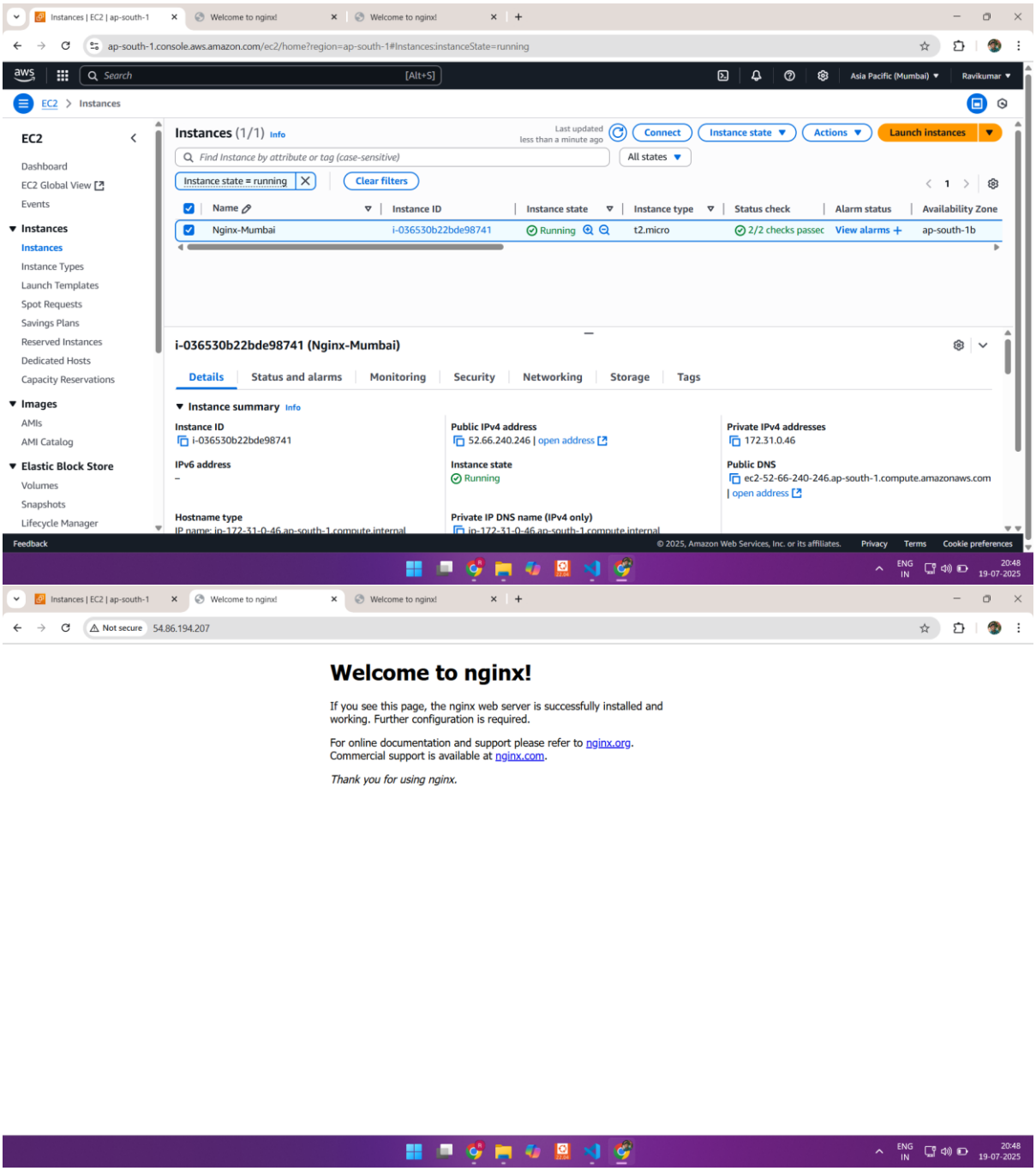

File Edit Selection View Go Run Terminal Help
Terraform_Task2
EXPLORER
TERRAFORM_TASK2
.history
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
terraform.tfvars
user_data.sh
variables.tf
main.tf
variables.tf
terraform.tfvars
user_data.sh
terraform.tfvars
key_name = "ec2-key"
regions = {
"ap-south-1" = {
ami = "ami-0f5ee92e2d63afc18" # Ubuntu/Mumbai
instance_type = "t2.micro"
},
"us-east-1" = {
ami = "ami-053b0d53c279acc90" # Ubuntu/Virginia
instance_type = "t2.micro"
}
}
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Ravikumar R@DESKTOP-JMSLBGI MINGW64 ~/Desktop/Terraform_Task2
$
powershell
bash
OUTLINE
TIMELINE
LOCAL HISTORY
Ln 1, Col 20 (3 selected) Spaces: 4 UTF-8 CRLF Plain Text
ENG IN
20:49
19-07-2025

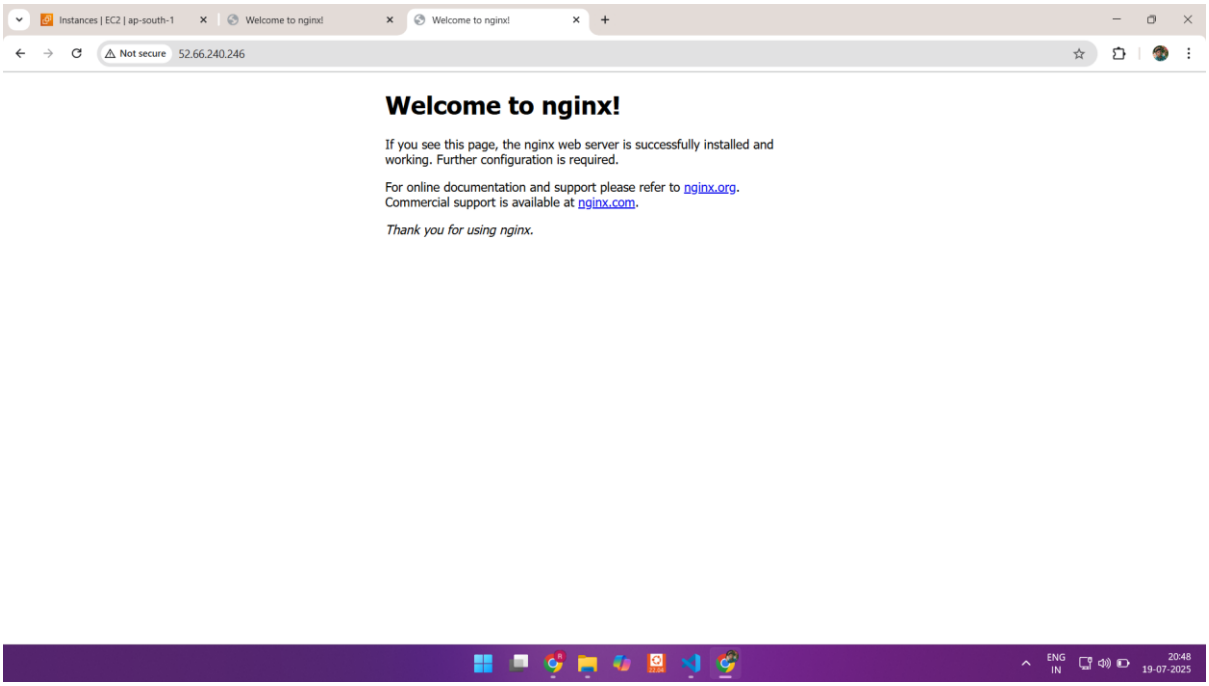

File Edit Selection View Go Run Terminal Help
Terraform_Task2
EXPLORER
TERRAFORM_TASK2
.history
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
terraform.tfvars
user_data.sh
variables.tf
main.tf
variables.tf
terraform.tfvars
user_data.sh
user_data.sh
1 #!/bin/bash
2 sudo apt update -y
3 sudo apt install nginx -y
4 sudo systemctl start nginx
5
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Ravikumar R@DESKTOP-JMSLBGI MINGW64 ~/Desktop/Terraform_Task2
$
powershell
bash
OUTLINE
TIMELINE
LOCAL HISTORY
Ln 5, Col 1 Spaces: 4 UTF-8 LF {} Shell Script
ENG IN
20:49
19-07-2025


File Edit Selection View Go Run Terminal Help
Terraform_Task2
EXPLORER
TERRAFORM_TASK2
.history
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
terraform.tfvars
user_data.sh
variables.tf
main.tf
variables.tf
terraform.tfvars
user_data.sh
user_data.sh
1 #!/bin/bash
2 sudo apt update -y
3 sudo apt install nginx -y
4 sudo systemctl start nginx
5
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Ravikumar R@DESKTOP-JMSLBGI MINGW64 ~/Desktop/Terraform_Task2
$ terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Reusing previous version of hashicorp/template from the dependency lock file
- Using previously-installed hashicorp/aws v5.100.0
- Using previously-installed hashicorp/template v2.2.0
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.
Ravikumar R@DESKTOP-JMSLBGI MINGW64 ~/Desktop/Terraform_Task2
$ terraform plan
data.template_file.nginx_script: Reading...
data.template_file.nginx_script: Read complete after 0s [id=4e878be8445df6f390d758085b27d8db06696c1f6cce7134385befd96a4c6112]
aws_instance.ec2_ap_south_1: Refreshing state... [id=i-022e68ae03562344b]
aws_security_group.allow_http: Refreshing state... [id=sg-0f835ea99f0ae7e91]
powershell
bash
OUTLINE
TIMELINE
LOCAL HISTORY
Ln 5, Col 1 Spaces: 4 UTF-8 LF {} Shell Script
ENG IN
20:50
19-07-2025


File Edit Selection View Go Run Terminal Help
Terraform_Task2
EXPLORER
TERRAFORM_TASK2
.history
.terraform
.terraform.lock.hcl
main.tf
terraform.tfstate
terraform.tfstate.backup
terraform.tfvars
user_data.sh
variables.tf
main.tf
variables.tf
terraform.tfvars
user_data.sh
$ user_data.sh
1 #!/bin/bash
2 sudo apt update -y
3 sudo apt install nginx -y
4 sudo systemctl start nginx
5
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
bash
+ revoke_rules_on_delete = false
+ tags_all = (known after apply)
+ vpc_id = (known after apply)
}
Plan: 4 to add, 0 to change, 3 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.
Ravikumar R@DESKTOP-JM6LBGI MINGW64 ~/Desktop/Terraform_Task2
$ terraform apply
data.template_file.nginx_script: Reading...
data.template_file.nginx_script: Read complete after 0s [id=4e878be8445df6f390d758085b27d8db06696c1f6cce7134385befd96a4c6112]
aws_instance.ec2_us_east_1: Refreshing state... [id=i-052eb6426384f1c41]
aws_instance.ec2_ap_south_1: Refreshing state... [id=i-022e68ae0356234b]
aws_security_group.allow_http: Refreshing state... [id=sg-0f835ea99f0ae7e91]
Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create
- destroy
-/+ destroy and then create replacement
Terraform will perform the following actions:
OUTLINE
TIMELINE
LOCAL HISTORY
Ln 5, Col 1 Spaces: 4 UTF-8 LF {} Shell Script
ENG IN 20:50 19-07-2025


Instances | EC2 | us-east-1
Welcome to nginx!
Welcome to nginx!
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Instances:instanceState=running
aws
Search [Alt+S]
United States (N. Virginia)
Ravikumar
EC2
EC2 > Instances
EC2
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Instances (1/1) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Instance state = running
Clear filters
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public I
Nginx-Virginia
i-083e542c17a013d9d
Running
t2.micro
2/2 checks passec
View alarms +
us-east-1d
ec2-54-
i-083e542c17a013d9d (Nginx-Virginia)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-083e542c17a013d9d
IPv6 address
–
Hostname type
Public IPv4 address
54.86.194.207 | open address
Instance state
Running
Private IP DNS name (IPv4 only)
Private IPv4 addresses
172.31.84.26
Public DNS
ec2-54-86-194-207.compute-1.amazonaws.com |
open address
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
ENG IN 20:48 19-07-2025


Instances | EC2 | ap-south-1
Welcome to nginx!
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:instanceState=running
Search [Alt+S]
Asia Pacific (Mumbai)
Ravikumar
EC2 > Instances
EC2
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Instances (1/1) Info
Last updated less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Instance state = running
Clear filters
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Nginx-Mumbai
i-036530b22bde98741
Running
t2.micro
2/2 checks passed
View alarms
ap-south-1b
i-036530b22bde98741 (Nginx-Mumbai)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-036530b22bde98741
IPv6 address
Hostname type
Public IPv4 address
52.66.240.246 | open address
Instance state
Running
Private IP DNS name (IPv4 only)
Private IPv4 addresses
172.31.0.46
Public DNS
ec2-52-66-240-246.ap-south-1.compute.amazonaws.com | open address
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
ENG IN
20:48
19-07-2025
Not secure
54.86.194.207
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.


Instances | EC2 | ap-south-1
Welcome to nginx!
Welcome to nginx!
Not secure
52.66.240.246
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.
ENG
IN
20:48
19-07-2025