ami



= var AMTS[var REGTON]

resource "aws_instance" "exercise2" {





















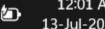


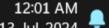














0

```
MINGW64:/d/terraform-examples/exercise2
                                                                                                                                      0
Star@Dell-I3 MINGW64 /d/terraform-examples/exercise2
$ cat provider.tf
provider "aws" {
 region = var.REGION
Star@Dell-I3 MINGW64 /d/terraform-examples/exercise2
$ cat var.tf
variable "REGION" {
 default = "us-east-1"
variable "ZONE1" {
 default = "us-east-2a"
variable "AMIS" {
 type = map(any)
 default = {
   us-east-2 = "ami-03657b56516ab7912"
   us-east-1 = "ami-0b72821e2f351e396"
Star@Dell-I3 MINGW64 /d/terraform-examples/exercise2
$ terraform plan
aws_instance.exercise2: Refreshing state... [id=i-08a63ba97ab68d450]
Terraform used the selected providers to generate the following execution
plan. Resource actions are indicated with the following symbols:
 + create
Terraform will perform the following actions:
 # aws_instance.exercise2 will be created
 + resource "aws_instance" "exercise2" {
                                              = "ami-0b72821e2f351e396"
     + ami
                                              = (known after apply)
      + arn
     + associate_public_ip_address
                                              = (known after apply)
     + availability_zone
                                              = (known after apply)
                                              = (known after apply)
      + cpu_core_count
                                              = (known after apply)
      + cpu_threads_per_core
                                              = (known after apply)
      + disable_api_stop
       disable ani termination
```



















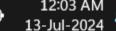














Note: You didn't use the -out option to save this plan, so Terraform can't quarantee to take exactly these actions if you run "terraform apply" now.

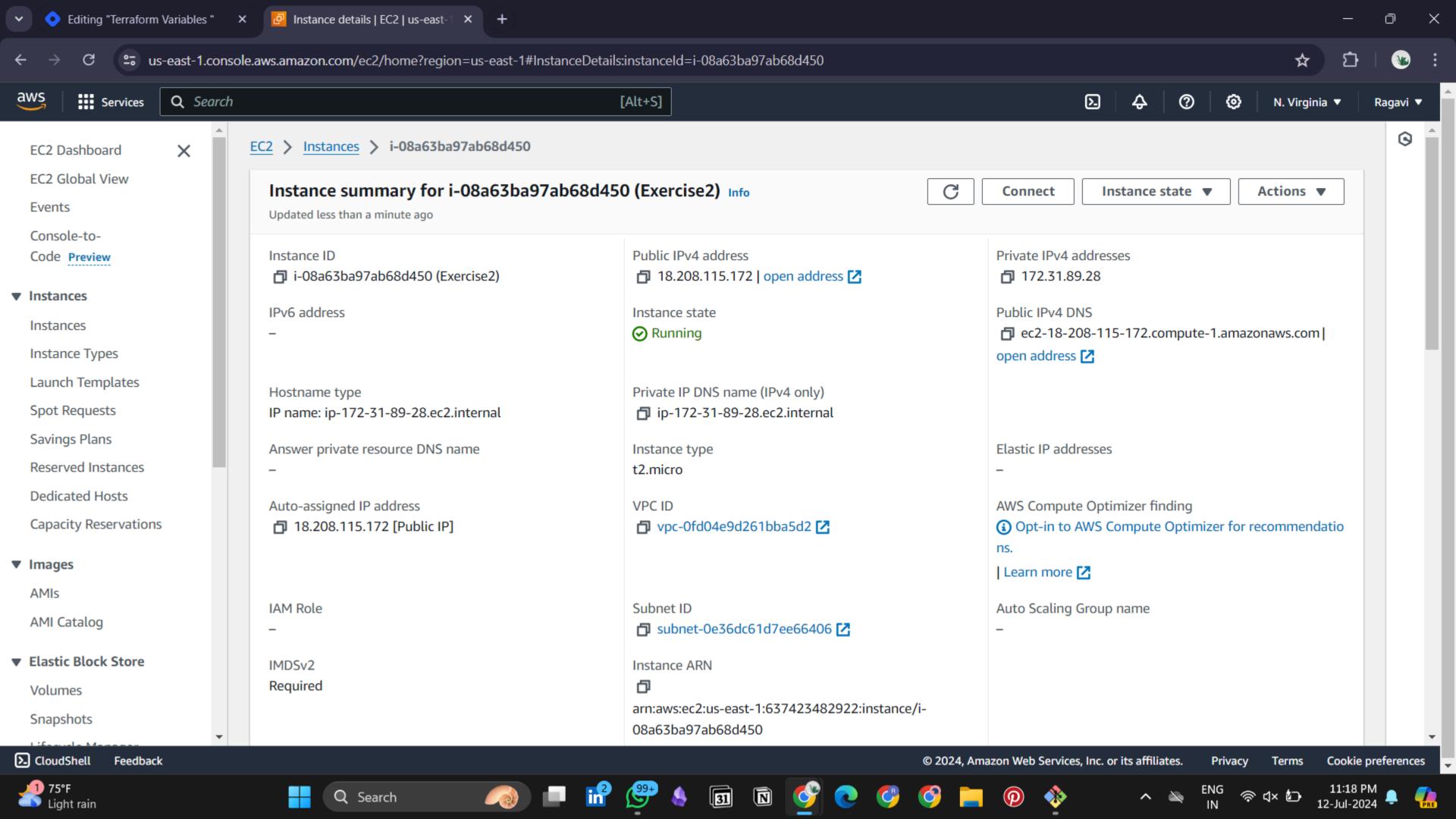
```
Star@Dell-I3 MINGW64 /d/terraform-examples/exercise2
$ terraform apply
aws_instance.exercise2: Refreshing state... [id=i-08a63ba97ab68d450]
```

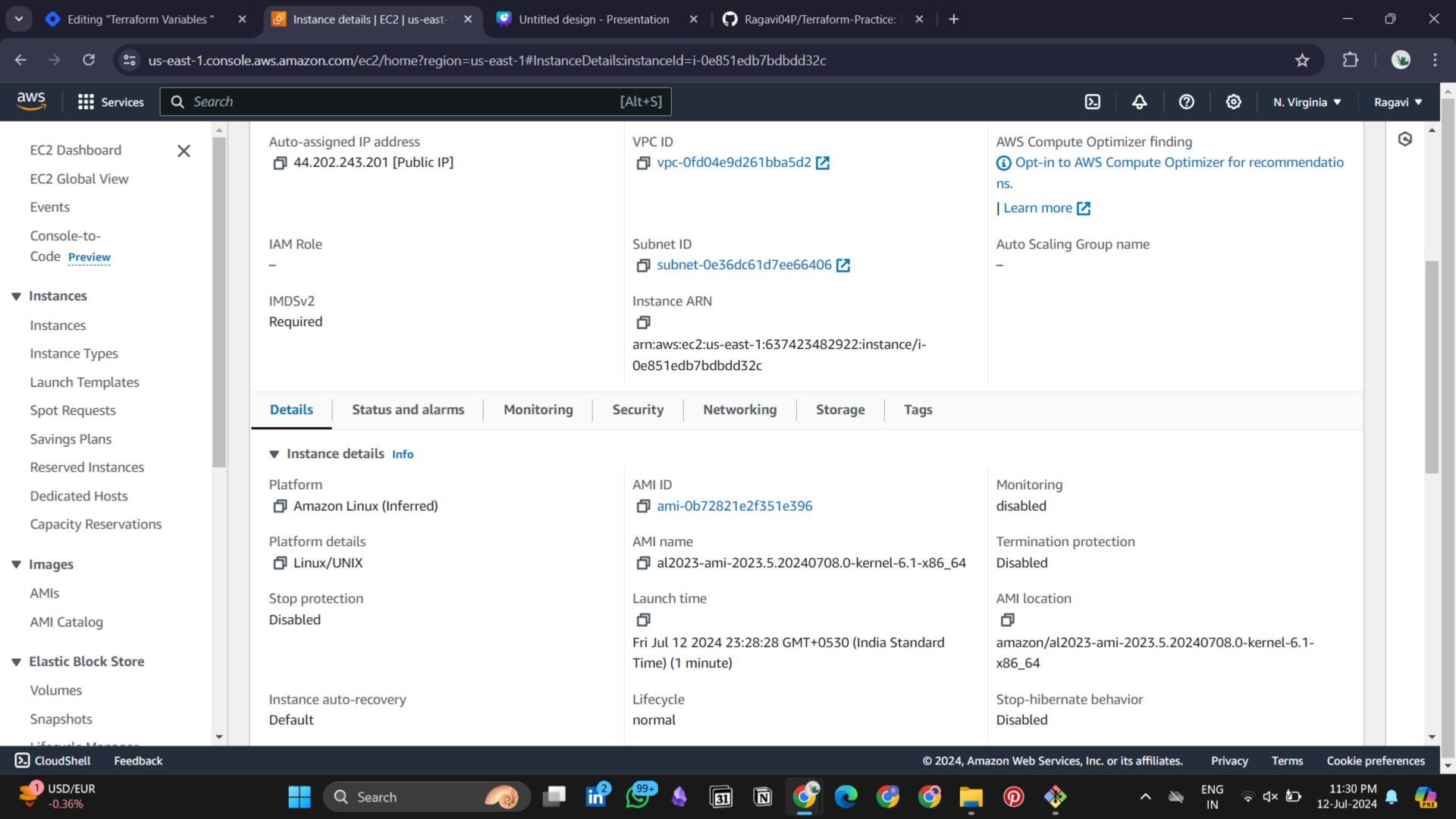
Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols: + create

Terraform will perform the following actions:

Plan: 1 to add, 0 to change, 0 to destroy.

```
# aws_instance.exercise2 will be created
+ resource "aws_instance" "exercise2" {
                                            = "ami-0b72821e2f351e396"
    + ami
                                            = (known after apply)
    + arn
                                            = (known after apply)
    + associate_public_ip_address
    + availability_zone
                                            = (known after apply)
                                            = (known after apply)
    + cpu_core_count
    + cpu_threads_per_core
                                            = (known after apply)
                                            = (known after apply)
    + disable_api_stop
    + disable_api_termination
                                            = (known after apply)
     ebs_optimized
                                            = (known after apply)
                                            = false
    + get_password_data
    + host_id
                                            = (known after apply)
                                            = (known after apply)
    + host_resource_group_arn
    + iam_instance_profile
                                            = (known after apply)
                                           = (known after apply)
    + id
    + instance_initiated_shutdown_behavior = (known after apply)
    + instance_lifecycle
                                            = (known after apply)
                                            = (known after apply)
     instance_state
                                            = "t2.micro"
     instance_type
                                            = (known after apply)
    + ipv6_address_count
                                            = (known after annly)
      inv6 addresses
```





```
NINGW64:/d/terraform-examples/exercise2
                                                                                                                                     0
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
Star@Dell-I3 MINGW64 /d/terraform-examples/exercise2
$ vim var.tf
Star@Dell-I3 MINGW64 /d/terraform-examples/exercise2
$ cat var.tf
variable "REGION" {
 default = "us-east-1"
variable "ZONE1" {
 default = "us-east-2a"
variable "AMIS" {
 type = map(any)
 default = {
   us-east-2 = "ami-03657b56516ab7912"
   us-east-1 = "ami-04a81a99f5ec58529"
Star@Dell-I3 MINGW64 /d/terraform-examples/exercise2
$ terraform plan
aws_instance.exercise2: Refreshing state... [id=i-0e851edb7bdbdd32c]
Terraform used the selected providers to generate the following execution
plan. Resource actions are indicated with the following symbols:
-/+ destroy and then create replacement
Terraform will perform the following actions:
 # aws_instance.exercise2 must be replaced
-/+ resource "aws_instance" "exercise2" {
                                              = "ami-0b72821e2f351e396" -> "ami-04a81a99f5ec58529" # forces replacement
     ~ ami
                                              = "arn:aws:ec2:us-east-1:637423482922:instance/i-0e851edb7bdbdd32c" -> (known after apply)
     ~ arn
     ~ associate_public_ip_address
                                              = true -> (known after apply)
     ~ availability_zone
                                             = "us-east-1c" -> (known after apply)
                                             = 1 -> (known after apply)
     ~ cpu_core_count
                                             = 1 -> (known after apply)
     ~ cpu_threads_per_core
     ~ disable ani ston
                                              = false -> (known after annly)
                           Q Search
```

```
NINGW64:/d/terraform-examples/exercise2
                                                                                                                                     0
Plan: 1 to add, 0 to change, 1 to destroy.
Note: You didn't use the -out option to save this plan, so Terraform can't
quarantee to take exactly these actions if you run "terraform apply" now.
Star@Dell-I3 MINGW64 /d/terraform-examples/exercise2
$ terraform apply
aws_instance.exercise2: Refreshing state... [id=i-0e851edb7bdbdd32c]
Terraform used the selected providers to generate the following execution
plan. Resource actions are indicated with the following symbols:
-/+ destroy and then create replacement
Terraform will perform the following actions:
 # aws_instance.exercise2 must be replaced
-/+ resource "aws_instance" "exercise2" {
                                             = "ami-0b72821e2f351e396" -> "ami-04a81a99f5ec58529" # forces replacement
     ~ ami
                                             = "arn:aws:ec2:us-east-1:637423482922:instance/i-0e851edb7bdbdd32c" -> (known after apply)
     ~ arn
     ~ associate_public_ip_address
                                             = true -> (known after apply)
                                             = "us-east-1c" -> (known after apply)
     ~ availability_zone
     ~ cpu_core_count
                                             = 1 -> (known after apply)
     ~ cpu_threads_per_core
                                             = 1 -> (known after apply)
     ~ disable_api_stop
                                             = false -> (known after apply)
     ~ disable_api_termination
                                             = false -> (known after apply)
                                             = false -> (known after apply)
     ~ ebs_optimized
                                             = false -> null

    hibernation

     + host_id
                                             = (known after apply)
                                             = (known after apply)
     + host_resource_group_arn
                                             = (known after apply)
     + iam_instance_profile
                                             = "i-0e851edb7bdbdd32c" -> (known after apply)
     ~ id
     instance_initiated_shutdown_behavior = "stop" -> (known after apply)
     + instance_lifecycle
                                             = (known after apply)
                                             = "running" -> (known after apply)
     ~ instance_state
                                             = 0 -> (known after apply)
     ipv6_address_count
       inv6 addresses
                                             = [] - (known after annly)
                           Q Search
```

```
MINGW64:/d/terraform-examples/exercise2
                                                                                                                                     0
 Terraform will perform the actions described above.
 Only 'yes' will be accepted to approve.
 Enter a value: yes
aws_instance.exercise2: Destroying... [id=i-0e851edb7bdbdd32c]
aws_instance.exercise2: Still destroying... [id=i-0e851edb7bdbdd32c, 10s elapsed]
aws_instance.exercise2: Still destroying... [id=i-0e851edb7bdbdd32c, 20s elapsed]
aws_instance.exercise2: Still destroying... [id=i-0e851edb7bdbdd32c, 30s elapsed]
aws_instance.exercise2: Still destroying... [id=i-0e851edb7bdbdd32c, 40s elapsed]
aws_instance.exercise2: Destruction complete after 45s
aws_instance.exercise2: Creating...
aws_instance.exercise2: Still creating... [10s elapsed]
aws_instance.exercise2: Still creating... [20s elapsed]
aws_instance.exercise2: Creation complete after 26s [id=i-0bdd258658235d08a]
Apply complete! Resources: 1 added, 0 changed, 1 destroyed.
Star@Dell-I3 MINGW64 /d/terraform-examples/exercise2
$ terraform destroy
aws_instance.exercise2: Refreshing state... [id=i-0bdd258658235d08a]
Terraform used the selected providers to generate the following execution
plan. Resource actions are indicated with the following symbols:
destroy
Terraform will perform the following actions:
 # aws_instance.exercise2 will be destroyed
 - resource "aws_instance" "exercise2" {
                                             = "ami-04a81a99f5ec58529" -> null
     – ami
                                             = "arn:aws:ec2:us-east-1:637423482922:instance/i-0bdd258658235d08a" -> null
     arn
     associate_public_ip_address
                                             = true -> null
                                             = "us-east-1c" -> null
     availability_zone
     cpu_core_count
                                             = 1 \rightarrow null
                                             = 1 -> null
     cpu_threads_per_core
     disable_api_stop
                                             = false -> null

    disable_api_termination

                                             = false -> null
     ebs_optimized
                                             = false -> null
      - det nassword data
                                                                                                              へ N ENG 令 以× 加 12:05 AM 13-Jul-2024
                           Q Search
```

