School of Computer Science

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES DEHRADUN, UTTARAKHAND

System Provisioning and Configuration Management

Lab File (2022-2026)
6th Semester

Submitted To:

Dr. Hitesh Kumar Sharma

Submitted By:

Akshat Pandey
(500101788)
B Tech CSE
DevOps[6th Semester]
R2142220306
Batch - 1

EXPERIMENT 6

Lab Exercise: Terraform Variables Objective:

Learn how to define and use variables in Terraform configuration.

Prerequisites:

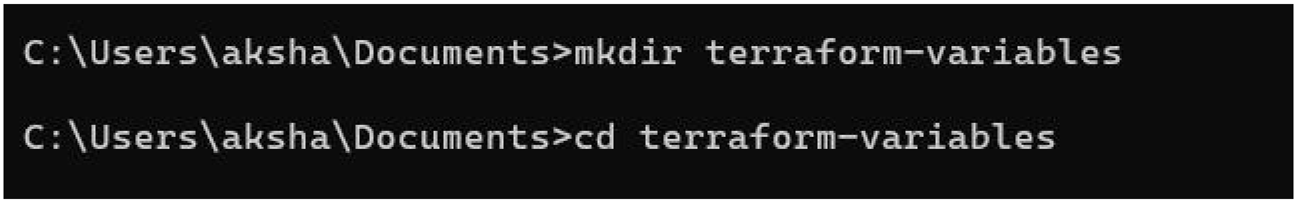
- Install Terraform on your machine.

Steps:

1. Create a Terraform Directory:

- Create a new directory for your Terraform project.

```
mkdir terraform-variables

cd terraform-variables
```



```
C:\Users\aksha\Documents>mkdir terraform-variables

C:\Users\aksha\Documents>cd terraform-variables
```

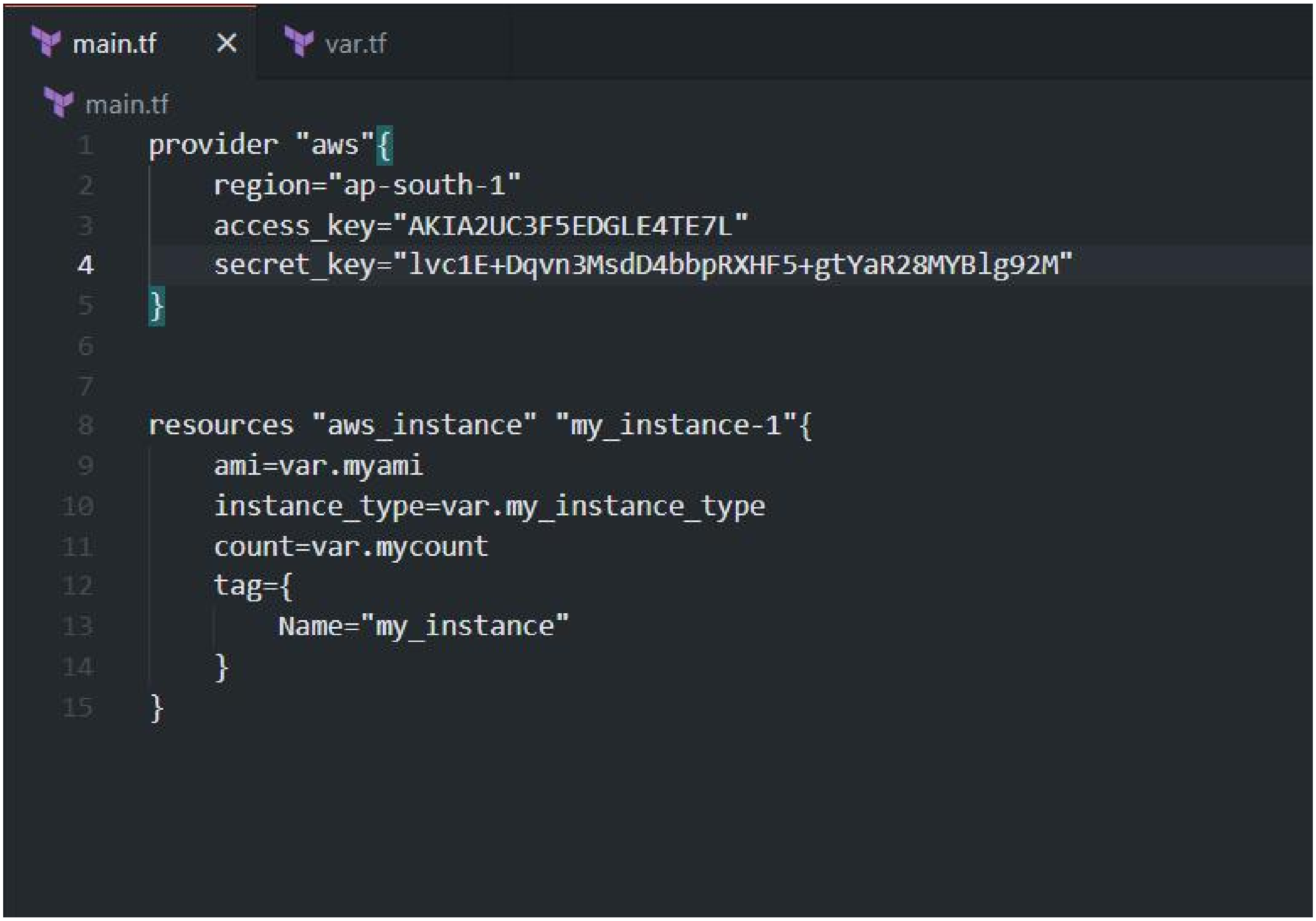
2. Create a Terraform Configuration File:

- Create a file named main.tf within your project directory.

# main.tf

```
resource "aws_instance" "myinstance-1" {
   ami = var.myami
   instance_type = var.my_instance_type
   count = var.mycount
   tags = {
     Name= "My Instance"
```

```
  }
}
```



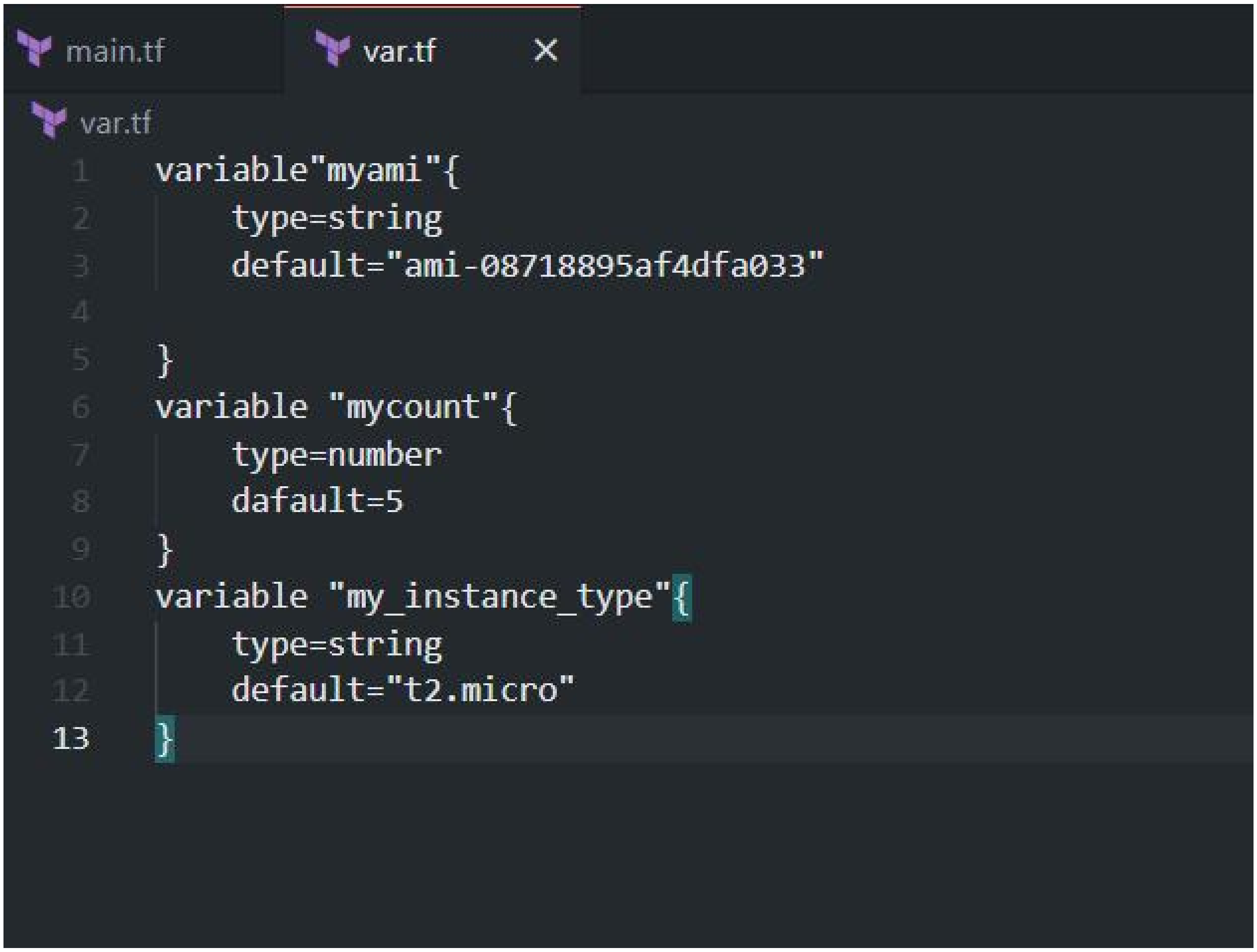
```
provider "aws"{
    region="ap-south-1"
    access_key="AKIA2UC3F5EDGLE4TE7L"
    secret_key="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"
}


resources "aws_instance" "my_instance-1"{
    ami=var.myami
    instance_type=var.my_instance_type
    count=var.mycount
    tag={
        Name="my_instance"
    }
}
```

3. Define Variables:

- Open a new file named variables.tf. Define variables for region, ami, and instance_type.

# variables.tf

```
variable "myami" {
  type = string
  default = "ami-08718895af4dfa033"
}


variable "mycount" {
  type = number
  default = 5
}


variable "my_instance_type" {
   type = string
   default = "t2.micro"
}
```



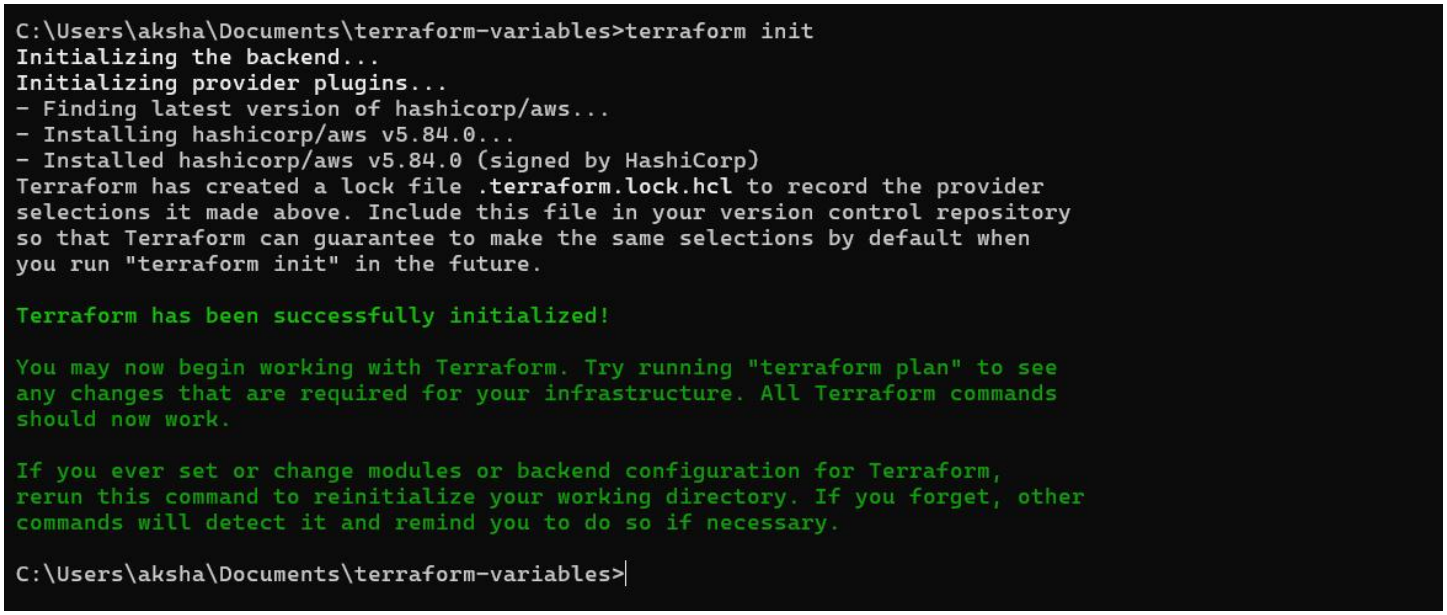
```
variable"myami"{
    type=string
    default="ami-08718895af4dfa033"

}
variable "mycount"{
    type=number
    dafault=5
}
variable "my_instance_type"{
    type=string
    default="t2.micro"
}
```

4. Initialize and Apply:

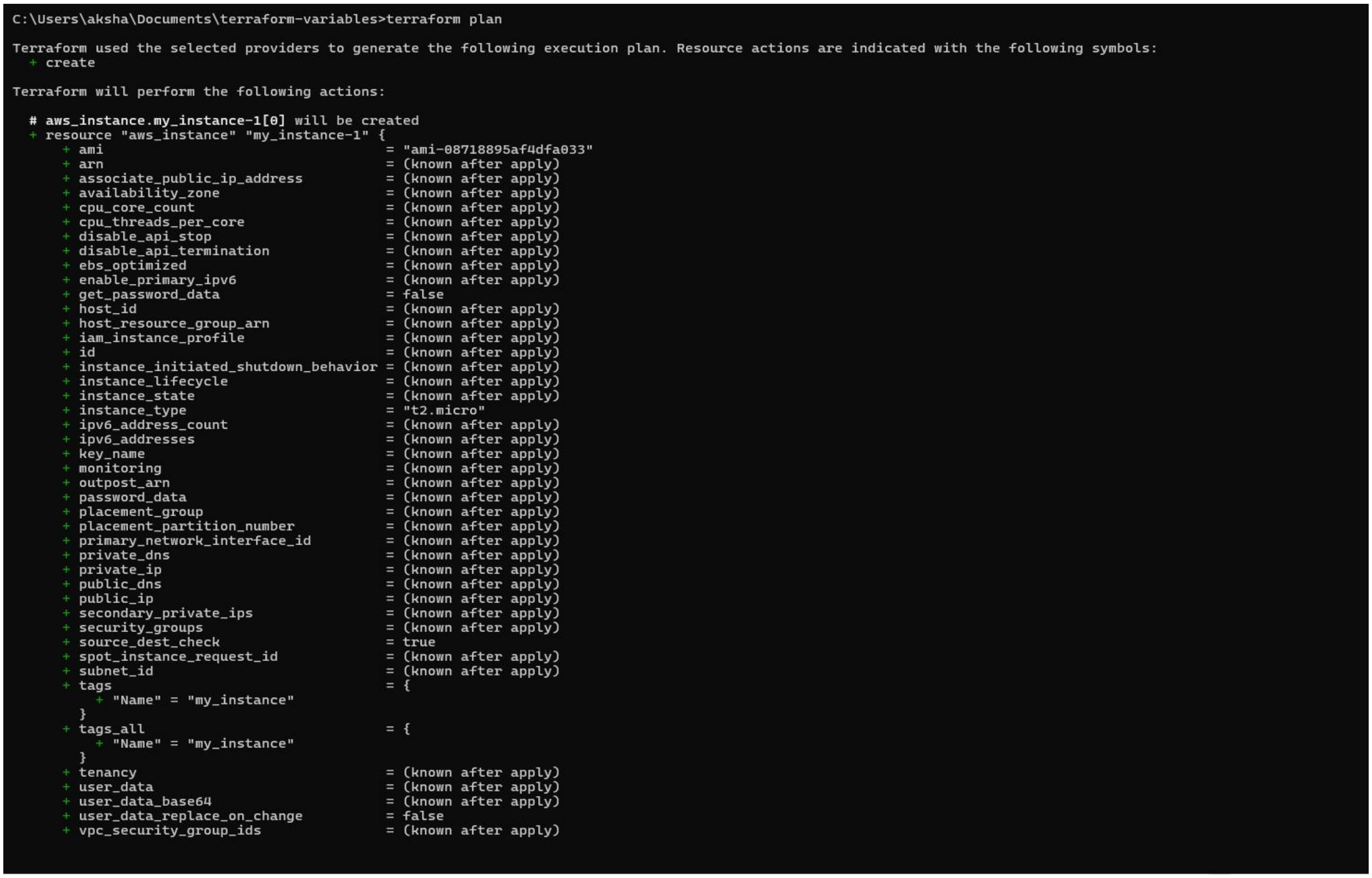
- Run the following Terraform commands to initialize and apply the configuration.

```
terraform init

terraform plan
terraform apply -auto-approve
```

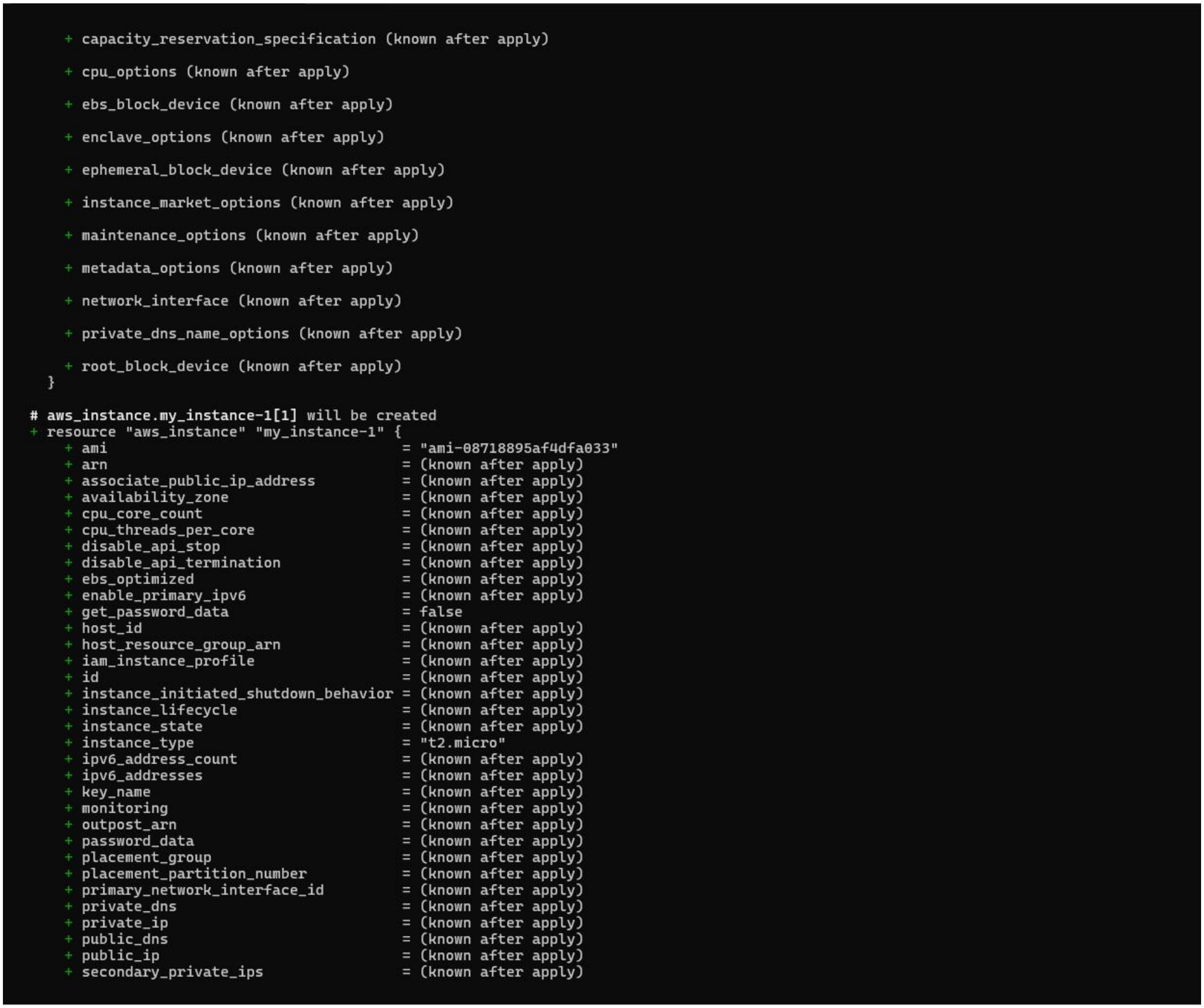
Observe how the region changes based on the variable override.



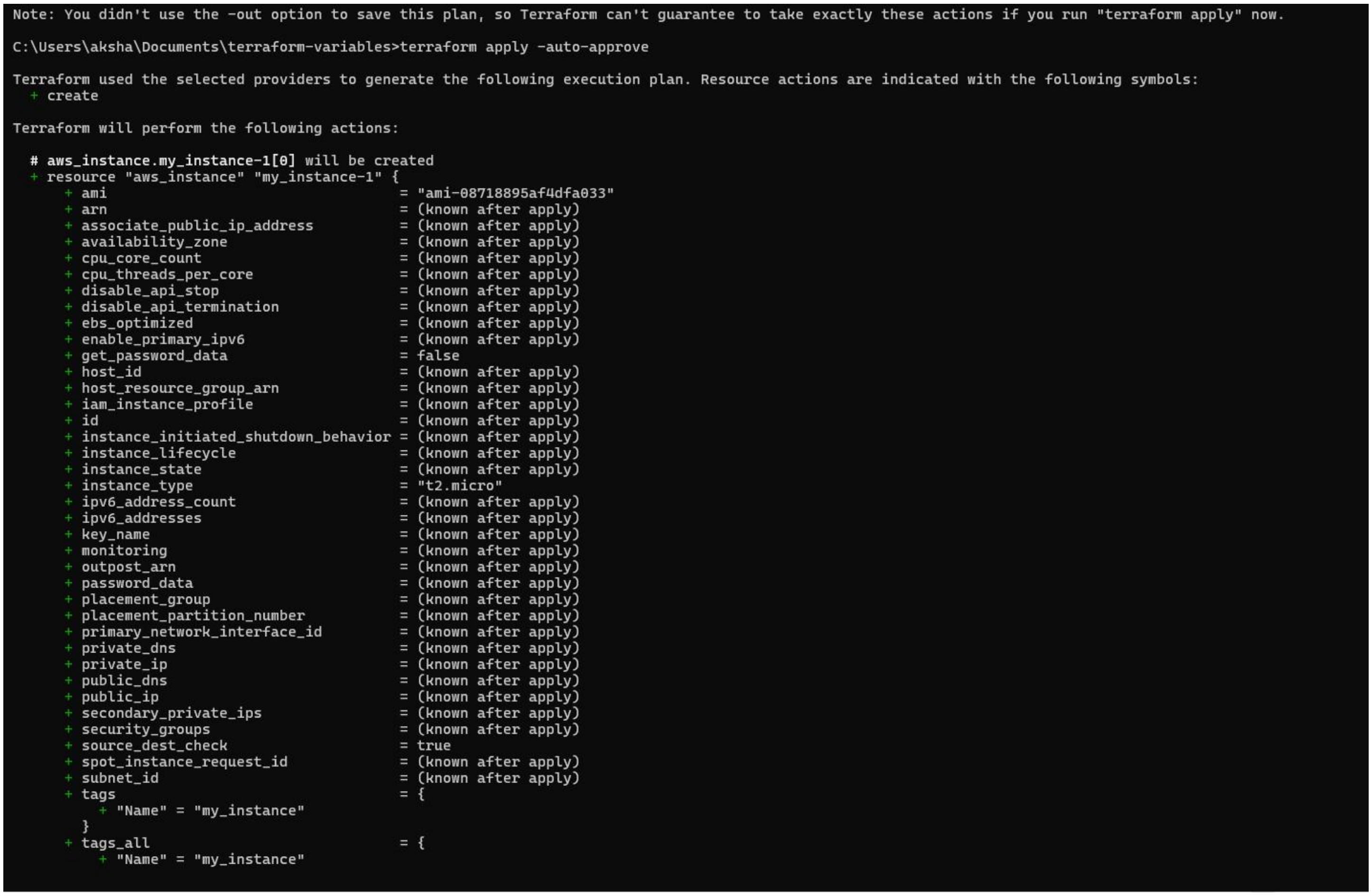
```
C:\Users\aksha\Documents\terraform-variables>terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.84.0...
- Installed hashicorp/aws v5.84.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl 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\aksha\Documents\terraform-variables>
```



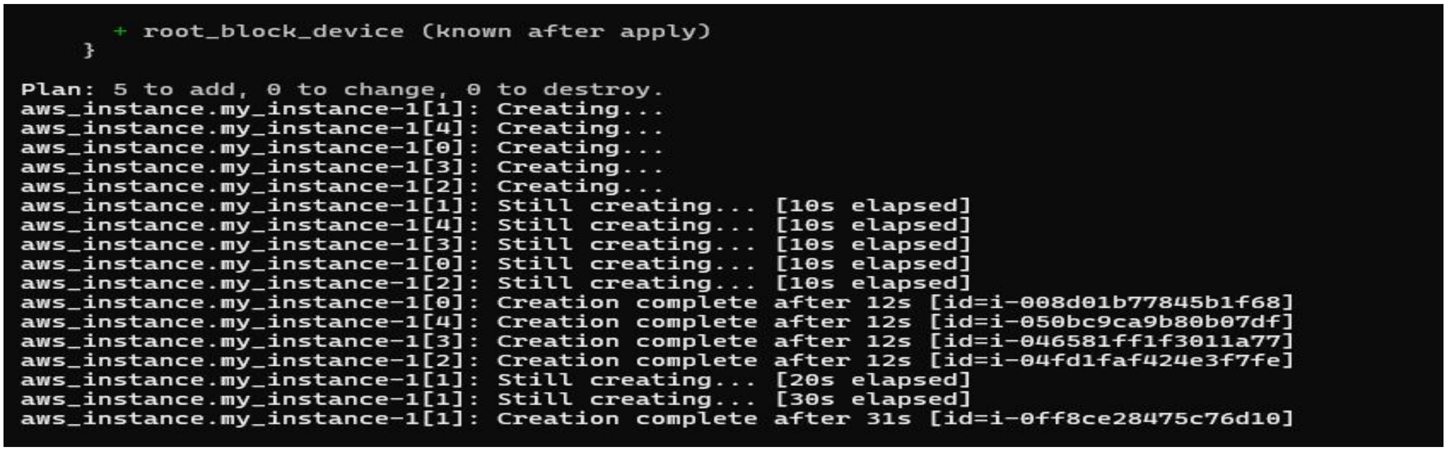
```
C:\Users\aksha\Documents\terraform-variables>terraform plan

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.my_instance-1[0] will be created
  + resource "aws_instance" "my_instance-1" {
      + ami                                  = "ami-08718895af4dfa033"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + enable_primary_ipv6                  = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = (known after apply)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
      + source_dest_check                    = true
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "my_instance"
        }
      + tags_all                             = {
          + "Name" = "my_instance"
        }
      + tenancy                              = (known after apply)
      + user_data                            = (known after apply)
      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = (known after apply)
```



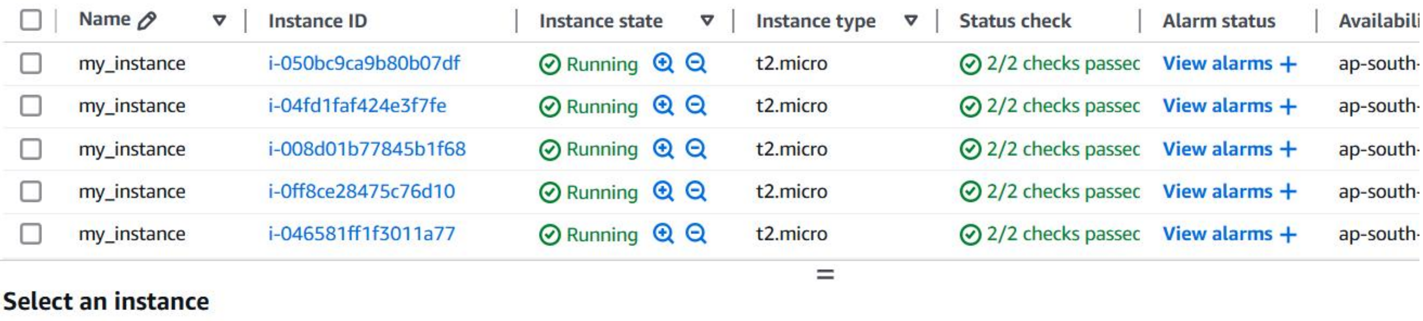
```
      + capacity_reservation_specification (known after apply)

      + cpu_options (known after apply)

      + 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)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

  # aws_instance.my_instance-1[1] will be created
  + resource "aws_instance" "my_instance-1" {
      + ami                                  = "ami-08718895af4dfa033"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + enable_primary_ipv6                  = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = (known after apply)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
```



```
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\aksha\Documents\terraform-variables>terraform apply -auto-approve

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.my_instance-1[0] will be created
  + resource "aws_instance" "my_instance-1" {
      + ami                                  = "ami-08718895af4dfa033"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + enable_primary_ipv6                  = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = (known after apply)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
      + source_dest_check                    = true
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "my_instance"
        }
      + tags_all                             = {
          + "Name" = "my_instance"
```



```
      + root_block_device (known after apply)
    }

Plan: 5 to add, 0 to change, 0 to destroy.
aws_instance.my_instance-1[1]: Creating...
aws_instance.my_instance-1[4]: Creating...
aws_instance.my_instance-1[0]: Creating...
aws_instance.my_instance-1[3]: Creating...
aws_instance.my_instance-1[2]: Creating...
aws_instance.my_instance-1[1]: Still creating... [10s elapsed]
aws_instance.my_instance-1[4]: Still creating... [10s elapsed]
aws_instance.my_instance-1[3]: Still creating... [10s elapsed]
aws_instance.my_instance-1[0]: Still creating... [10s elapsed]
aws_instance.my_instance-1[2]: Still creating... [10s elapsed]
aws_instance.my_instance-1[0]: Creation complete after 12s [id=i-008d01b77845b1f68]
aws_instance.my_instance-1[4]: Creation complete after 12s [id=i-050bc9ca9b80b07df]
aws_instance.my_instance-1[3]: Creation complete after 12s [id=i-046581ff1f3011a77]
aws_instance.my_instance-1[2]: Creation complete after 12s [id=i-04fd1faf424e3f7fe]
aws_instance.my_instance-1[1]: Still creating... [20s elapsed]
aws_instance.my_instance-1[1]: Still creating... [30s elapsed]
aws_instance.my_instance-1[1]: Creation complete after 31s [id=i-0ff8ce28475c76d10]
```



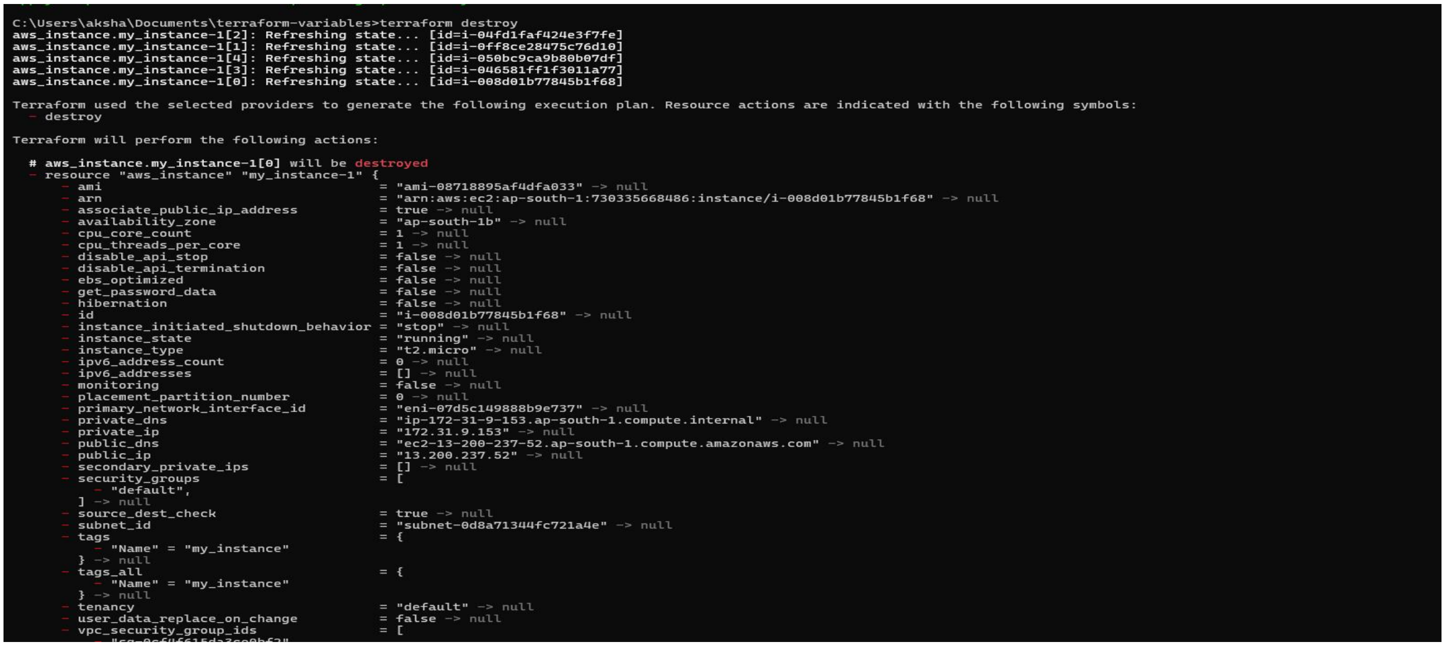
□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availabili
□	my_instance	i-050bc9ca9b80b07df	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	ap-south-
□	my_instance	i-04fd1faf424e3f7fe	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	ap-south-
□	my_instance	i-008d01b77845b1f68	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	ap-south-
□	my_instance	i-0ff8ce28475c76d10	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	ap-south-
□	my_instance	i-046581ff1f3011a77	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	ap-south-

Select an instance

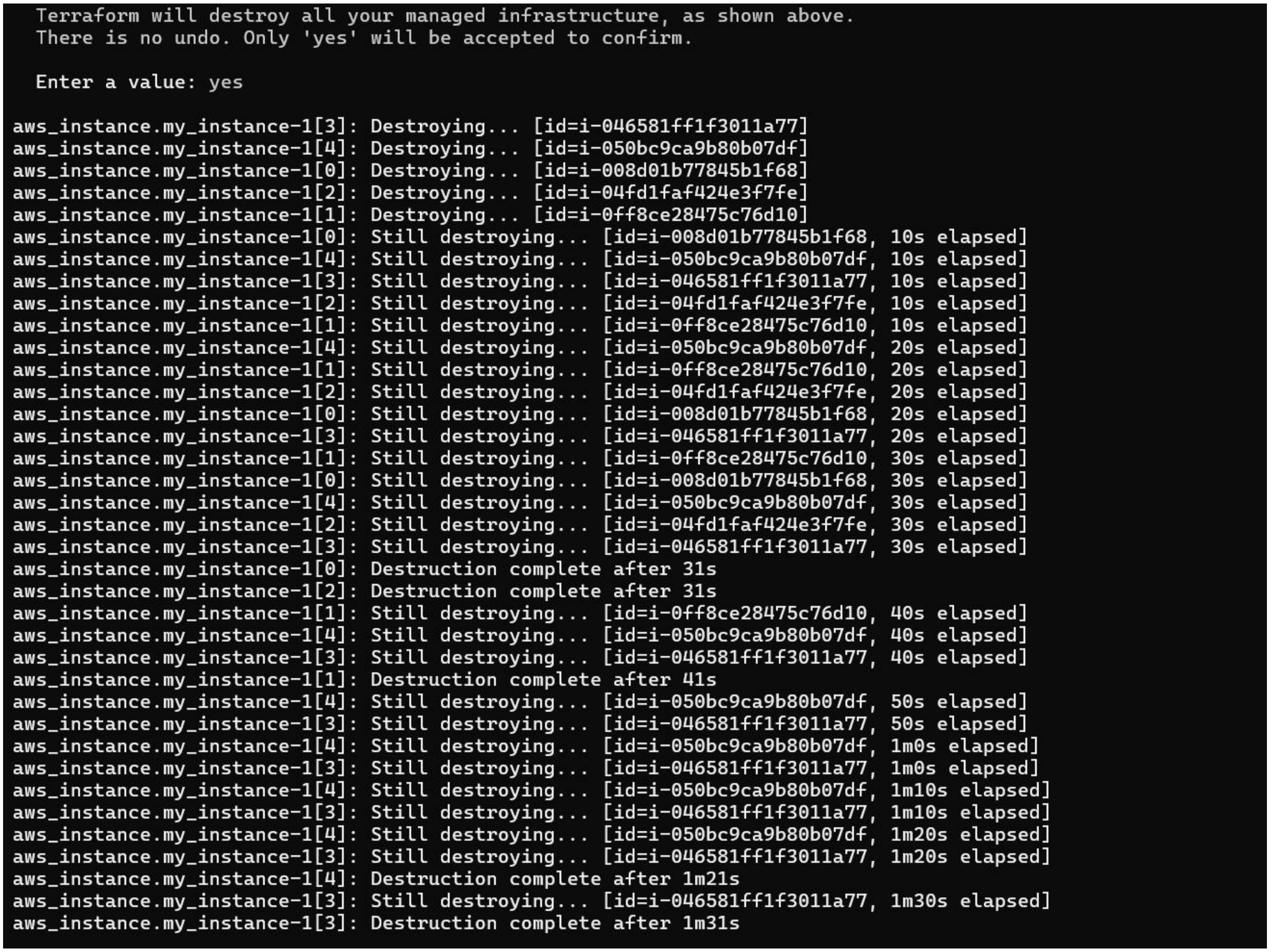
5. Clean Up:

After testing, you can clean up resources.

```
terraform destroy
```



```
C:\Users\aksha\Documents\terraform-variables>terraform destroy
aws_instance.my_instance-1[2]: Refreshing state... [id=i-04fd1faf424e3f7fe]
aws_instance.my_instance-1[1]: Refreshing state... [id=i-0ff8ce28475c76d10]
aws_instance.my_instance-1[4]: Refreshing state... [id=i-050bc9ca9b80b07df]
aws_instance.my_instance-1[3]: Refreshing state... [id=i-046581ff1f3011a77]
aws_instance.my_instance-1[0]: Refreshing state... [id=i-008d01b77845b1f68]

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.my_instance-1[0] will be destroyed
  - resource "aws_instance" "my_instance-1" {
      - ami                                  = "ami-08718895af4dfa033" -> null
      - arn                                  = "arn:aws:ec2:ap-south-1:730335668486:instance/i-008d01b77845b1f68" -> null
      - associate_public_ip_address          = true -> null
      - availability_zone                    = "ap-south-1b" -> null
      - cpu_core_count                       = 1 -> null
      - cpu_threads_per_core                 = 1 -> null
      - disable_api_stop                     = false -> null
      - disable_api_termination              = false -> null
      - ebs_optimized                        = false -> null
      - get_password_data                    = false -> null
      - hibernation                          = false -> null
      - id                                   = "i-008d01b77845b1f68" -> null
      - instance_initiated_shutdown_behavior = "stop" -> null
      - instance_state                       = "running" -> null
      - instance_type                        = "t2.micro" -> null
      - ipv6_address_count                   = 0 -> null
      - ipv6_addresses                       = [] -> null
      - monitoring                           = false -> null
      - placement_partition_number           = 0 -> null
      - primary_network_interface_id         = "eni-07d5c149888b9e737" -> null
      - private_dns                          = "ip-172-31-9-153.ap-south-1.compute.internal" -> null
      - private_ip                           = "172.31.9.153" -> null
      - public_dns                           = "ec2-13-200-237-52.ap-south-1.compute.amazonaws.com" -> null
      - public_ip                            = "13.200.237.52" -> null
      - secondary_private_ips                = [] -> null
      - security_groups                      = [
          - "default",
        ] -> null
      - source_dest_check                    = true -> null
      - subnet_id                            = "subnet-0d8a71344fc721a4e" -> null
      - tags                                 = {
          - "Name" = "my_instance"
        } -> null
      - tags_all                             = {
          - "Name" = "my_instance"
        } -> null
      - tenancy                              = "default" -> null
      - user_data_replace_on_change          = false -> null
      - vpc_security_group_ids               = [
          - "sg-0cf4f615da3ce0bf2"
```



```
  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.my_instance-1[3]: Destroying... [id=i-046581ff1f3011a77]
aws_instance.my_instance-1[4]: Destroying... [id=i-050bc9ca9b80b07df]
aws_instance.my_instance-1[0]: Destroying... [id=i-008d01b77845b1f68]
aws_instance.my_instance-1[2]: Destroying... [id=i-04fd1faf424e3f7fe]
aws_instance.my_instance-1[1]: Destroying... [id=i-0ff8ce28475c76d10]
aws_instance.my_instance-1[0]: Still destroying... [id=i-008d01b77845b1f68, 10s elapsed]
aws_instance.my_instance-1[4]: Still destroying... [id=i-050bc9ca9b80b07df, 10s elapsed]
aws_instance.my_instance-1[3]: Still destroying... [id=i-046581ff1f3011a77, 10s elapsed]
aws_instance.my_instance-1[2]: Still destroying... [id=i-04fd1faf424e3f7fe, 10s elapsed]
aws_instance.my_instance-1[1]: Still destroying... [id=i-0ff8ce28475c76d10, 10s elapsed]
aws_instance.my_instance-1[4]: Still destroying... [id=i-050bc9ca9b80b07df, 20s elapsed]
aws_instance.my_instance-1[1]: Still destroying... [id=i-0ff8ce28475c76d10, 20s elapsed]
aws_instance.my_instance-1[2]: Still destroying... [id=i-04fd1faf424e3f7fe, 20s elapsed]
aws_instance.my_instance-1[0]: Still destroying... [id=i-008d01b77845b1f68, 20s elapsed]
aws_instance.my_instance-1[3]: Still destroying... [id=i-046581ff1f3011a77, 20s elapsed]
aws_instance.my_instance-1[1]: Still destroying... [id=i-0ff8ce28475c76d10, 30s elapsed]
aws_instance.my_instance-1[0]: Still destroying... [id=i-008d01b77845b1f68, 30s elapsed]
aws_instance.my_instance-1[4]: Still destroying... [id=i-050bc9ca9b80b07df, 30s elapsed]
aws_instance.my_instance-1[2]: Still destroying... [id=i-04fd1faf424e3f7fe, 30s elapsed]
aws_instance.my_instance-1[3]: Still destroying... [id=i-046581ff1f3011a77, 30s elapsed]
aws_instance.my_instance-1[0]: Destruction complete after 31s
aws_instance.my_instance-1[2]: Destruction complete after 31s
aws_instance.my_instance-1[1]: Still destroying... [id=i-0ff8ce28475c76d10, 40s elapsed]
aws_instance.my_instance-1[4]: Still destroying... [id=i-050bc9ca9b80b07df, 40s elapsed]
aws_instance.my_instance-1[3]: Still destroying... [id=i-046581ff1f3011a77, 40s elapsed]
aws_instance.my_instance-1[1]: Destruction complete after 41s
aws_instance.my_instance-1[4]: Still destroying... [id=i-050bc9ca9b80b07df, 50s elapsed]
aws_instance.my_instance-1[3]: Still destroying... [id=i-046581ff1f3011a77, 50s elapsed]
aws_instance.my_instance-1[4]: Still destroying... [id=i-050bc9ca9b80b07df, 1m0s elapsed]
aws_instance.my_instance-1[3]: Still destroying... [id=i-046581ff1f3011a77, 1m0s elapsed]
aws_instance.my_instance-1[4]: Still destroying... [id=i-050bc9ca9b80b07df, 1m10s elapsed]
aws_instance.my_instance-1[3]: Still destroying... [id=i-046581ff1f3011a77, 1m10s elapsed]
aws_instance.my_instance-1[4]: Still destroying... [id=i-050bc9ca9b80b07df, 1m20s elapsed]
aws_instance.my_instance-1[3]: Still destroying... [id=i-046581ff1f3011a77, 1m20s elapsed]
aws_instance.my_instance-1[4]: Destruction complete after 1m21s
aws_instance.my_instance-1[3]: Still destroying... [id=i-046581ff1f3011a77, 1m30s elapsed]
aws_instance.my_instance-1[3]: Destruction complete after 1m31s
```

Confirm the destruction by typing yes.

6. Conclusion:

This lab exercise introduces you to Terraform variables and demonstrates how to use them in your configurations. Experiment with different variable values and overrides to understand their impact on the