School of Computer Science

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES
DEHRADUN, UTTARAKHAND

System Provisioning and Configuration Management

Lab File (2021-2025)
6th Semester

<u>*Submitted To:*</u>

Dr. Hitesh Kumar Sharma

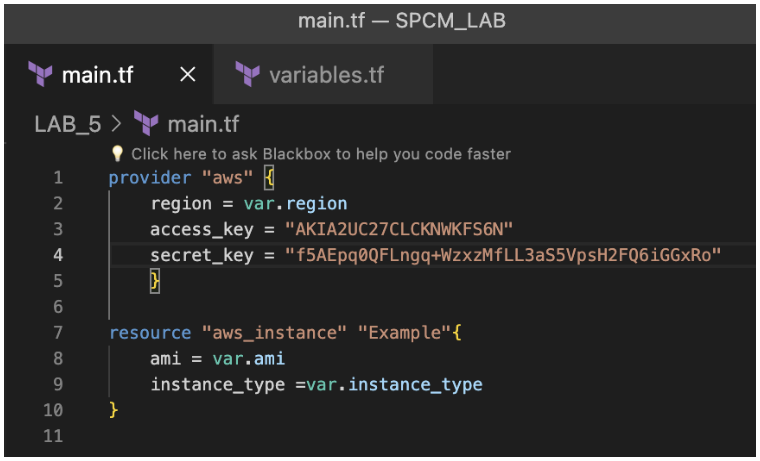
<u>*Submitted By:*</u>

Prakhar Gupta
(500093012)
B Tech CSE
DevOps[6th Semester]
R2142210572
Batch - 2

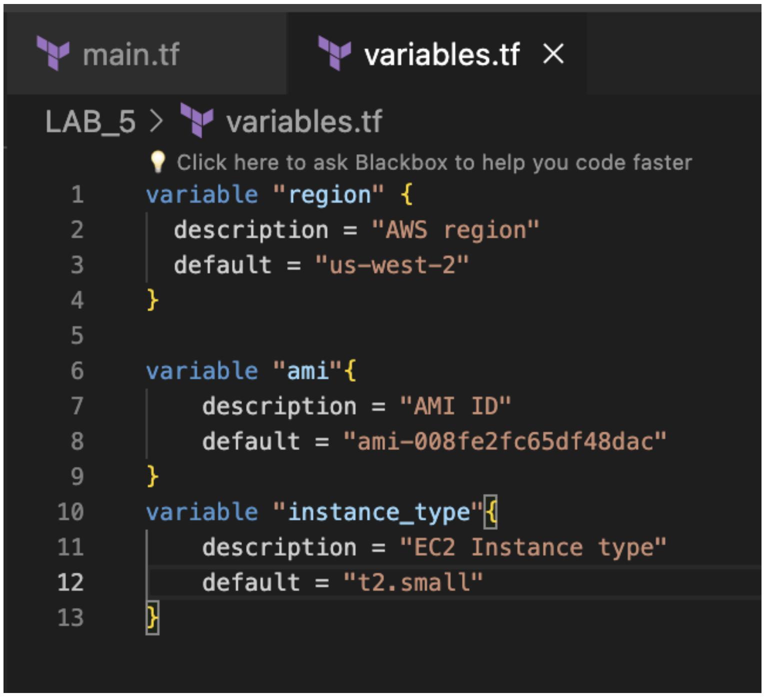
Lab Exercise 5– Terraform Variables with Command Line Arguments

. Use Command Line Arguments: • Open a terminal and navigate to your Terraform project directory. • Run the terraform init command:

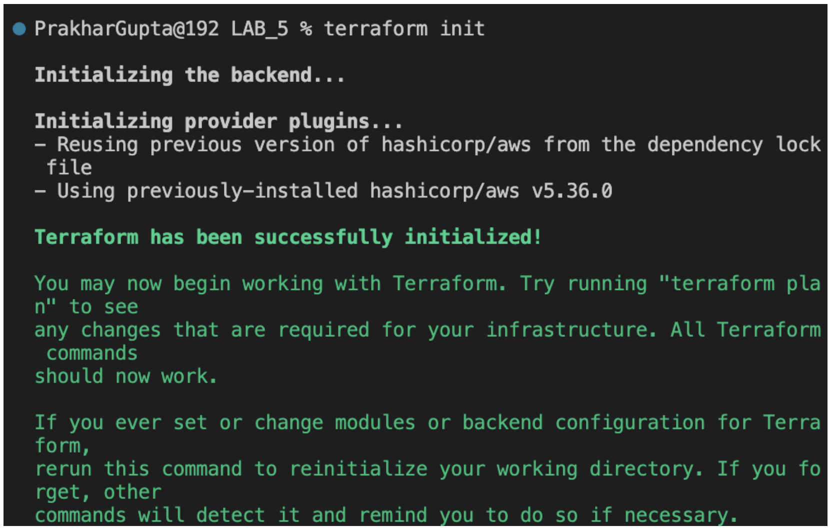
Run the terraform apply command with command line arguments to set variable values:



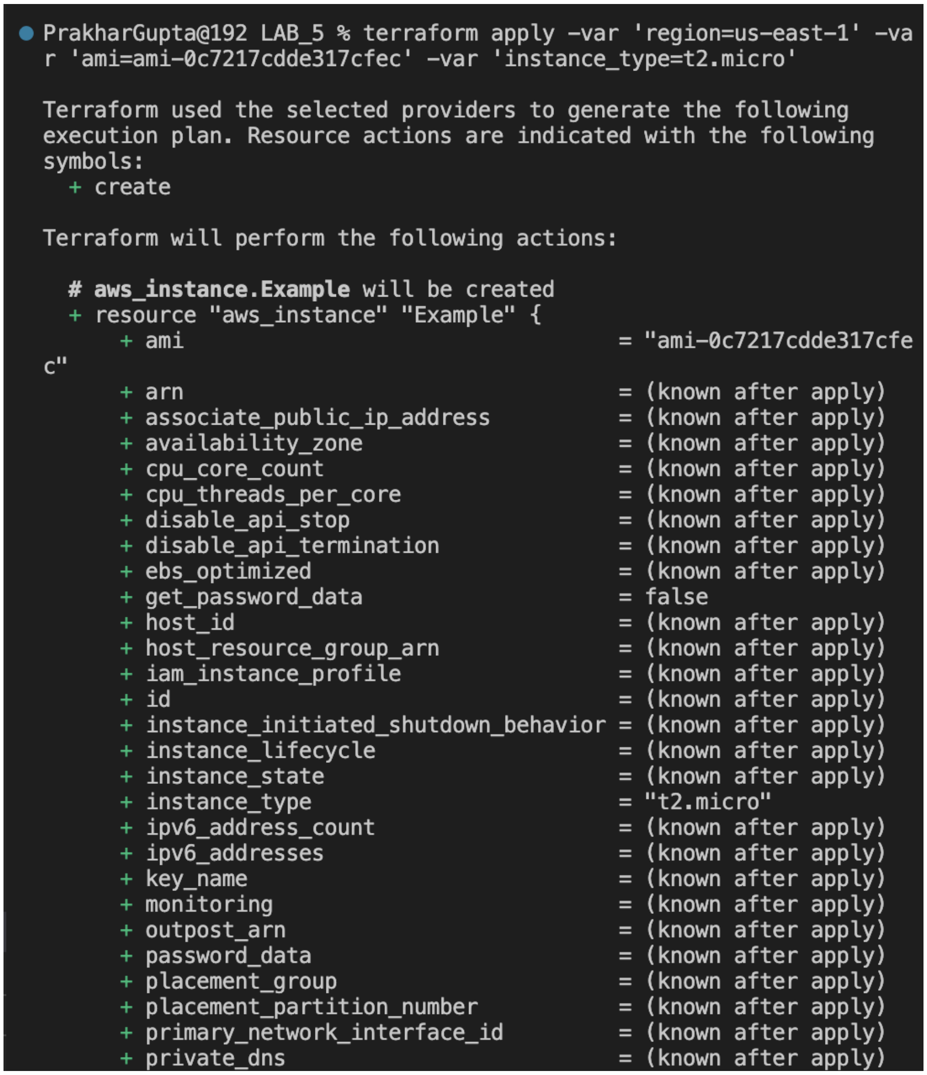
```
provider "aws" {
    region = var.region
    access_key = "AKIA2UC27CLCKNWKFS6N"
    secret_key = "f5AEpq0QFLngq+WzxzMfLL3aS5VpsH2FQ6iGGxRo"
    }

resource "aws_instance" "Example"{
    ami = var.ami
    instance_type =var.instance_type
}
```



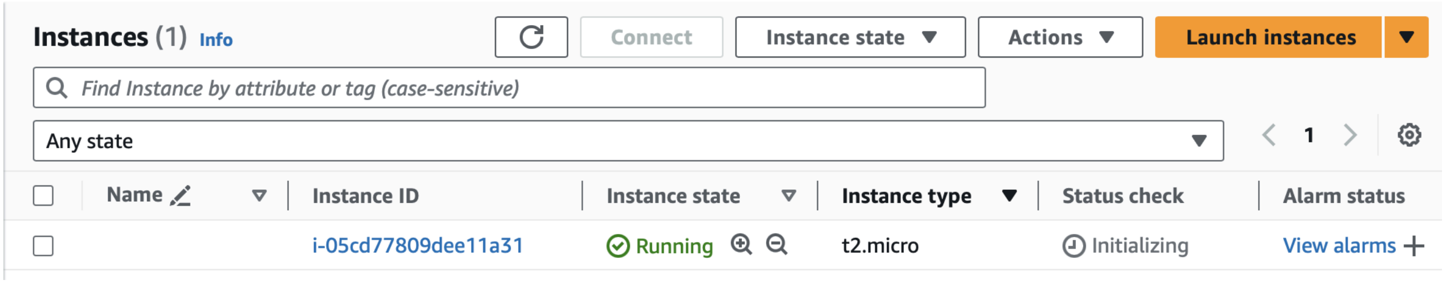
```
variable "region" {
  description = "AWS region"
  default = "us-west-2"
}

variable "ami"{
    description = "AMI ID"
    default = "ami-008fe2fc65df48dac"
}
variable "instance_type"{
    description = "EC2 Instance type"
    default = "t2.small"
}
```



```
● PrakharGupta@192 LAB_5 % terraform init

  Initializing the backend...

  Initializing provider plugins...
  - Reusing previous version of hashicorp/aws from the dependency lock
   file
  - Using previously-installed hashicorp/aws v5.36.0

  Terraform has been successfully initialized!

  You may now begin working with Terraform. Try running "terraform pla
  n" 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 Terra
  form,
  rerun this command to reinitialize your working directory. If you fo
  rget, other
  commands will detect it and remind you to do so if necessary.
```

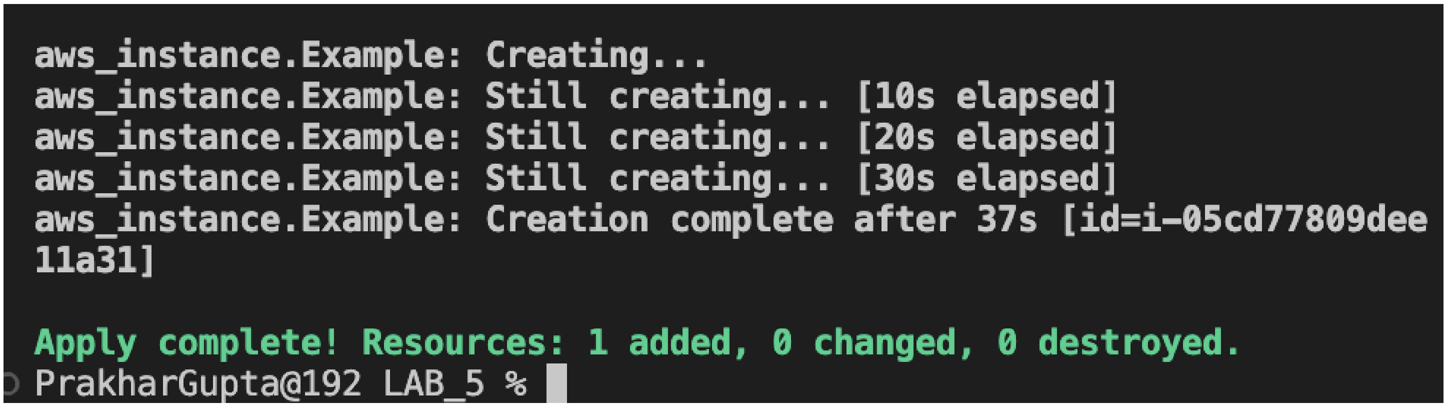


```
● PrakharGupta@192 LAB_5 % terraform apply -var 'region=us-east-1' -va
  r 'ami=ami-0c7217cdde317cfec' -var 'instance_type=t2.micro'

  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.Example will be created
    + resource "aws_instance" "Example" {
        + ami                                  = "ami-0c7217cdde317cfe
  c"
        + 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)
        + 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)
```



□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
□		i-05cd77809dee11a31	✓ Running	t2.micro	Initializing	View alarms +

Clean Up: After testing, you can clean up resources:



```
aws_instance.Example: Creating...
aws_instance.Example: Still creating... [10s elapsed]
aws_instance.Example: Still creating... [20s elapsed]
aws_instance.Example: Still creating... [30s elapsed]
aws_instance.Example: Creation complete after 37s [id=i-05cd77809dee
11a31]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
PrakharGupta@192 LAB_5 %
```