School of Computer Science

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES DEHRADUN, UTTARAKHAND

System Provisioning and Configuration Management

Lab File (2021-2025)
6th Semester

Submitted To:

Dr. Hitesh Kumar Sharma

Submitted By:

Antriksh Bahri
(500091315)
B Tech CSE
DevOps[6th Semester]
R2142210128
Batch - 1

Lab Exercise 1– Install Terraform on Windows

Installing Terraform on Windows requires you to download the correct Terraform package, unpack, and execute it via the CLI. Follow the instructions below to ensure you do not miss any steps.

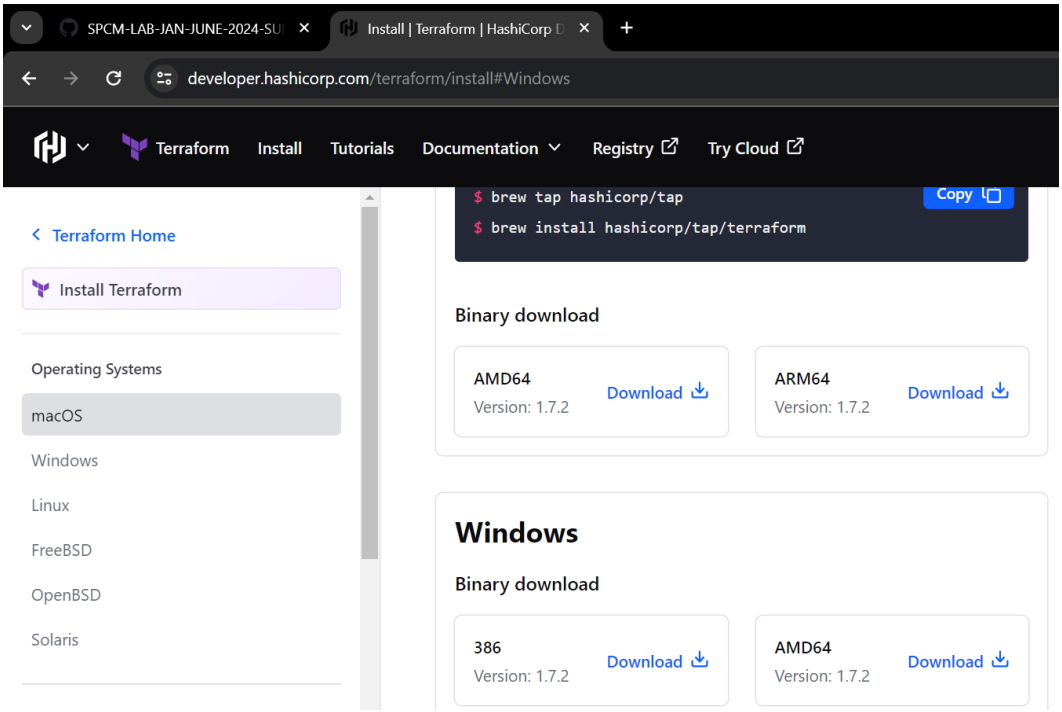
Download Terraform File for Windows

To find the latest version of Terraform for Windows:

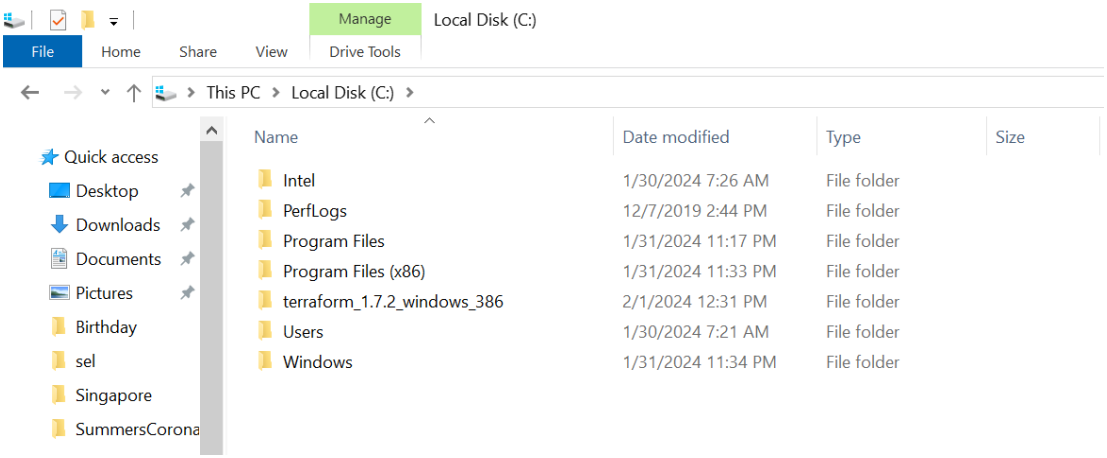
1. Browse to the Download Terraform page.

2. Select the Windows tab under the Operating System heading. The latest version is preselected.

3. Choose the binary to download. Select 386 for 32-bit systems or AMD64 for 64-bit systems. Choose the download location for the zip file if the download does not start automatically.



4. Unzip the downloaded file. For example, use the C:\terraform path. Remember this location so you can add the path to the environment variables.



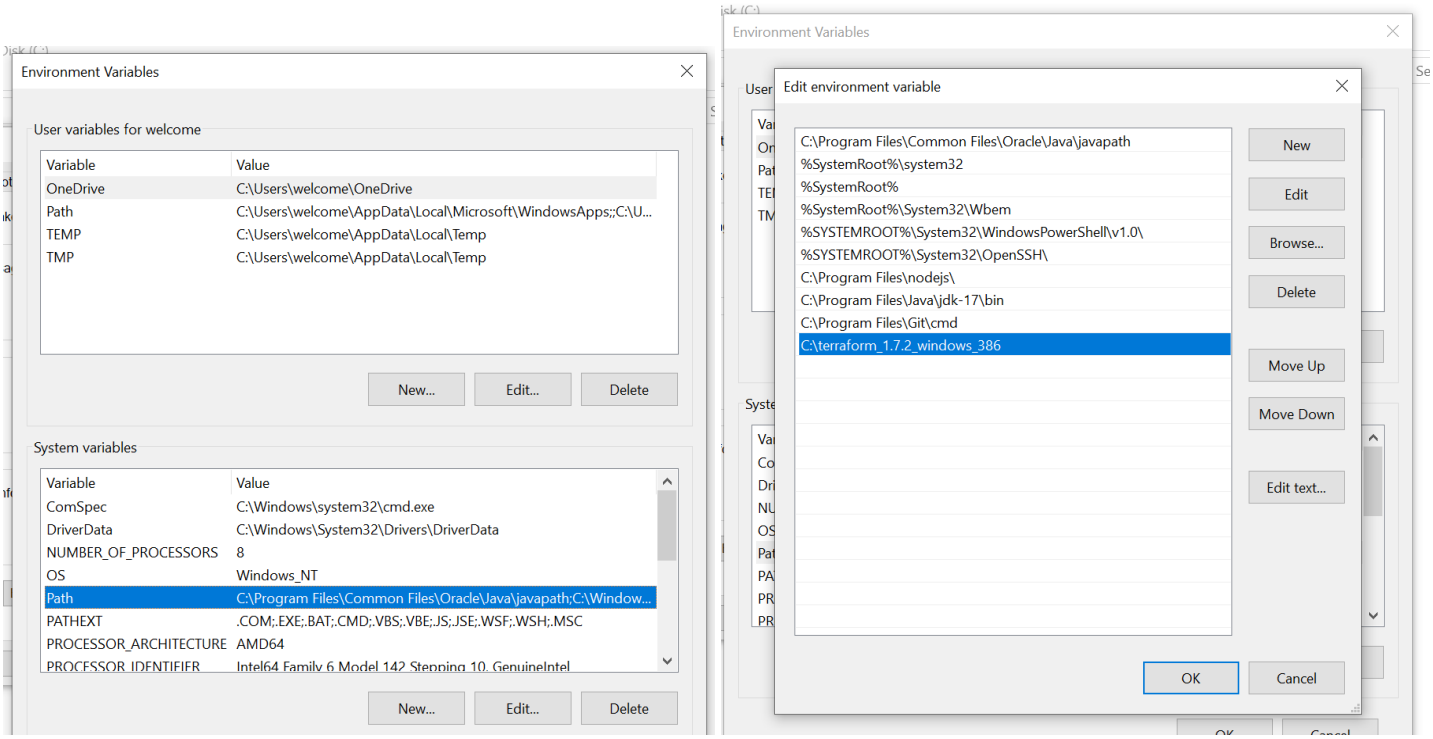
Add Terraform Path to System Environment Variables To add the Terraform executable to the system's global path:

1. Open the start menu, start typing environment and click Edit system environment variables. The System Properties window opens.

2. Click the Environment Variables... button.

3. Select the Path variable in the System variables section to add terraform for all accounts. Alternatively, select Path in the User variables section to add terraform for the currently logged-in user only. Click Edit once you select a Path.

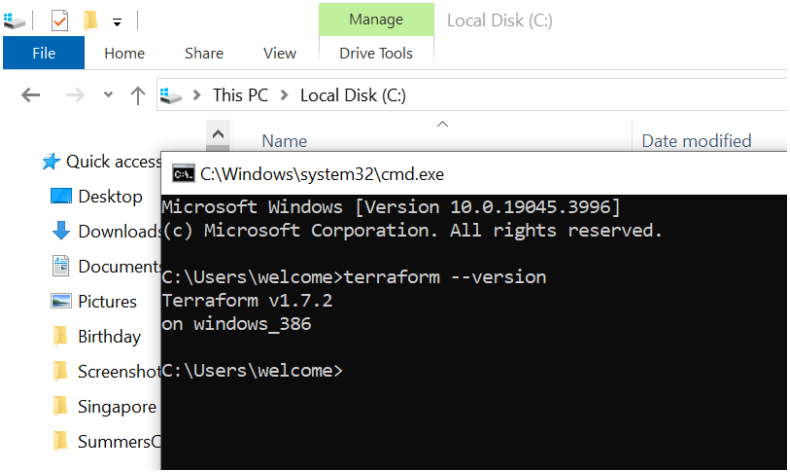
4. Click New in the edit window and enter the location of the Terraform folder.



Verify Windows Terraform Installation To check the Terraform global path configuration:

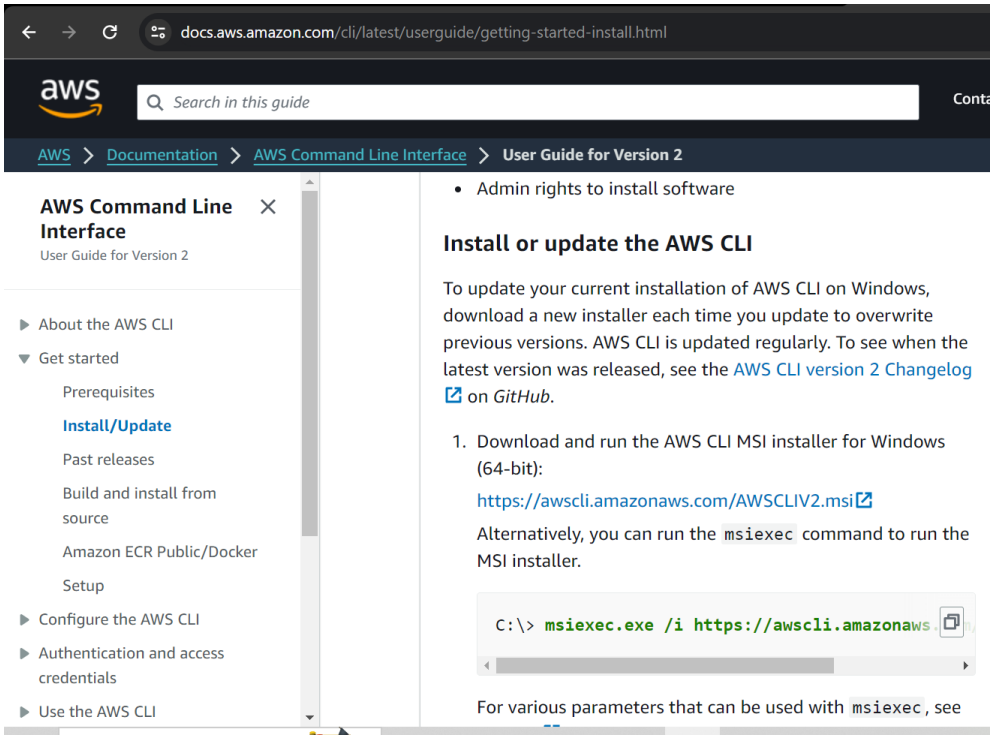
1. Open a new command-prompt window.

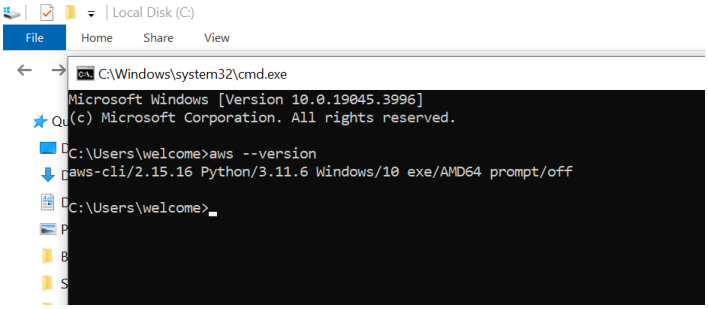
2. Enter the command to check the Terraform version: terraform -version

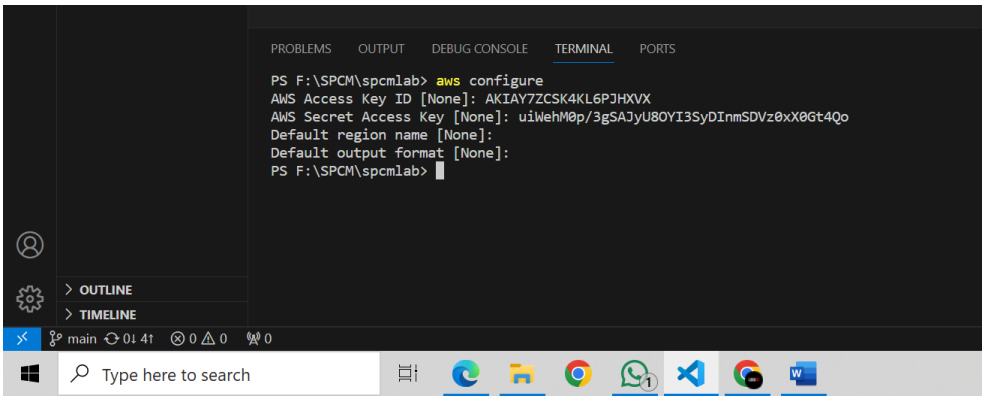


Lab Exercise 2– Terraform AWS Provider and IAM User Setting

AWS Credentials: Ensure you have AWS credentials (Access Key ID and Secret Access Key) configured. You can set them up using the AWS CLI or by setting environment variables.







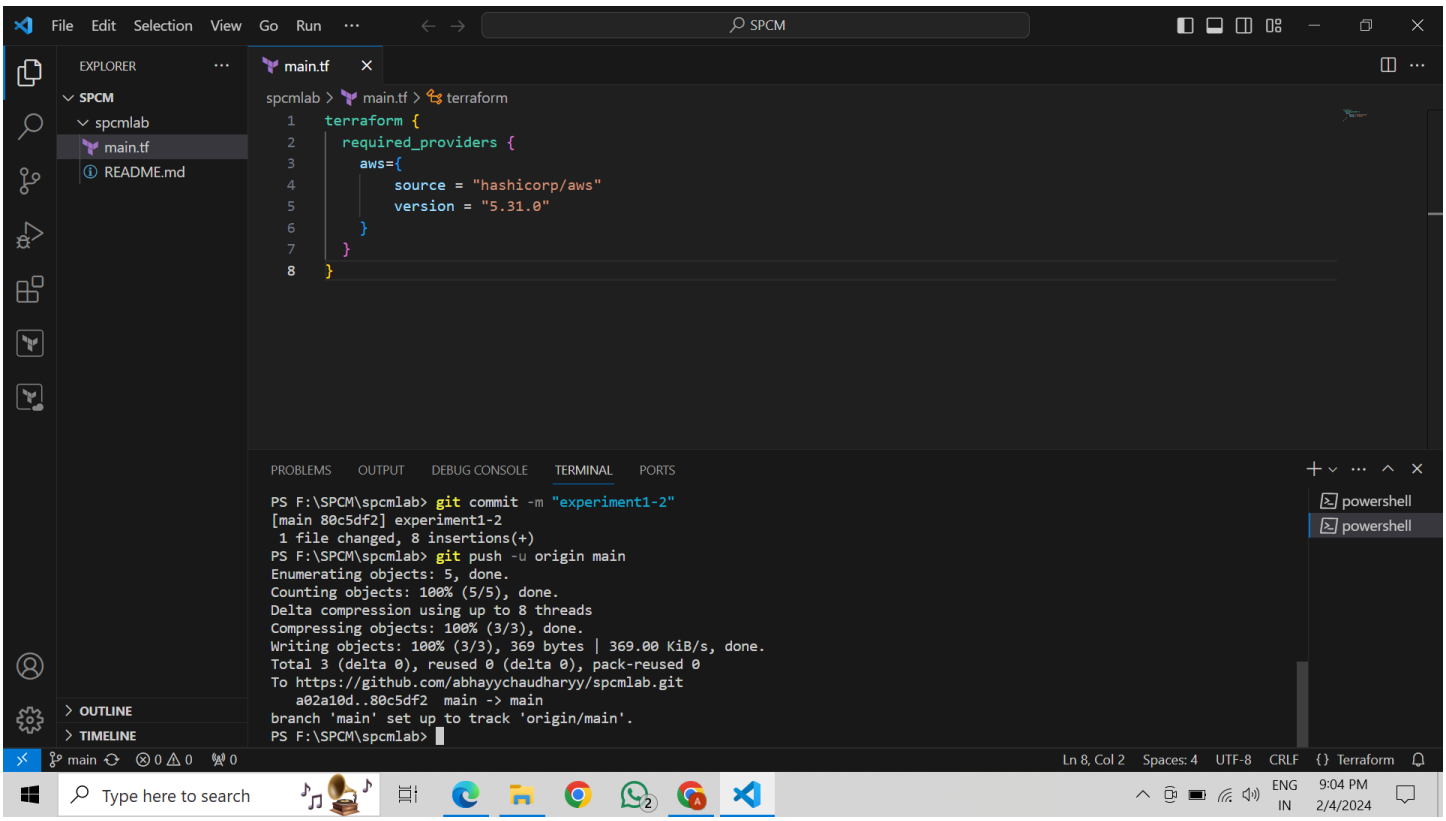
Exercise Steps: Step

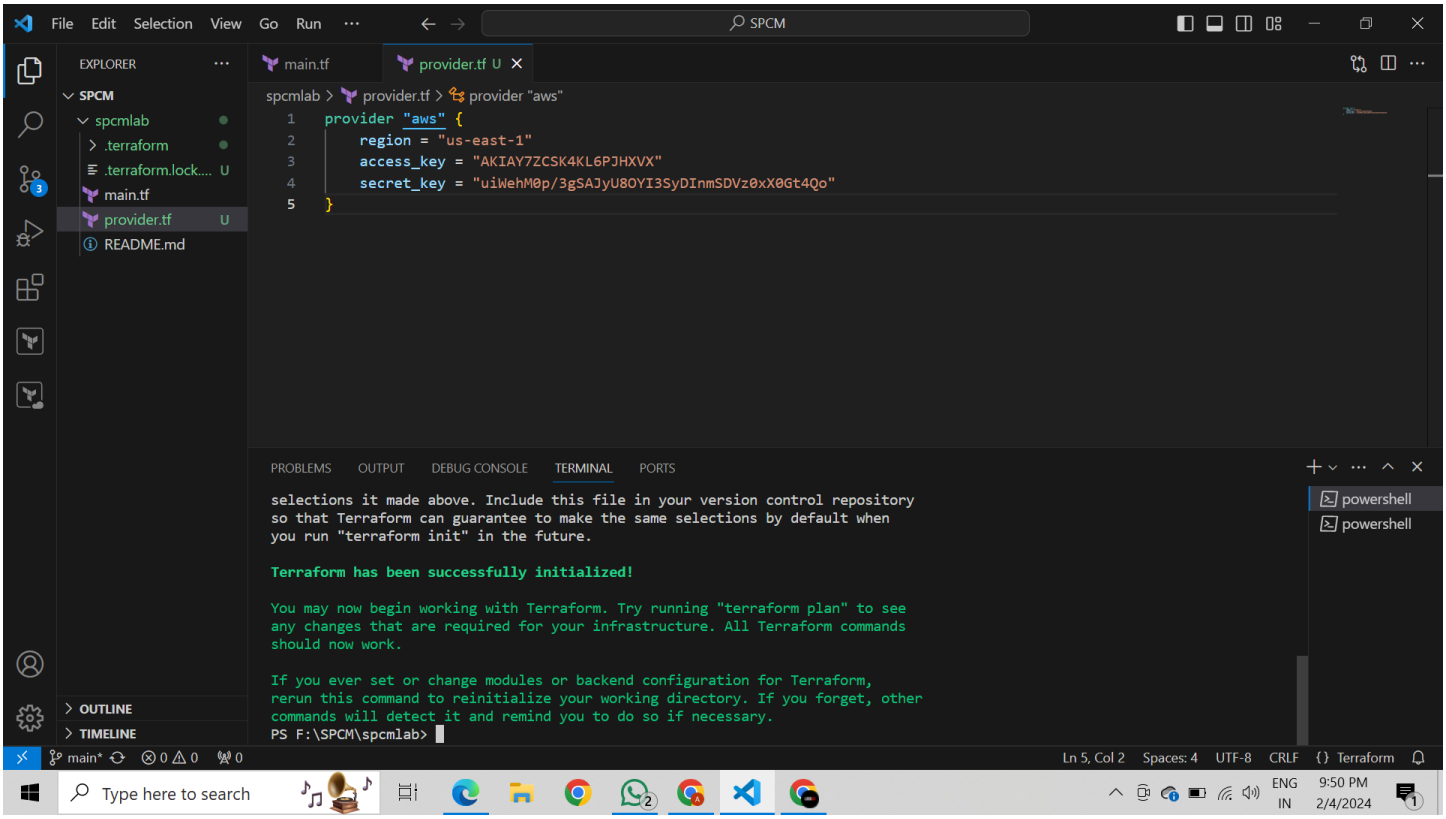
1: Create a New Directory: Create a new directory for your Terraform configuration:

2: Create Terraform Configuration File (main.tf): Create a file named main.tf with the following content:

This script defines an AWS provider and provisions an EC2 instance.

3: Initialize Terraform: Run the following command to initialize your Terraform working directory:





Lab Exercise 3–Provisioning an EC2 Instance on AWS

Exercise Steps:

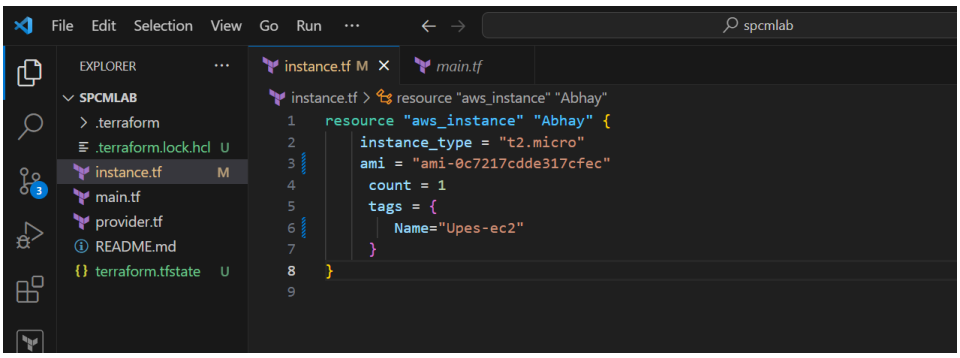
1: Create a New Directory: Create a new directory for your Terraform configuration:

2: Create Terraform Configuration File (main.tf): Create a file named main.tf with the following content:

This script defines an AWS provider and provisions an EC2 instance.

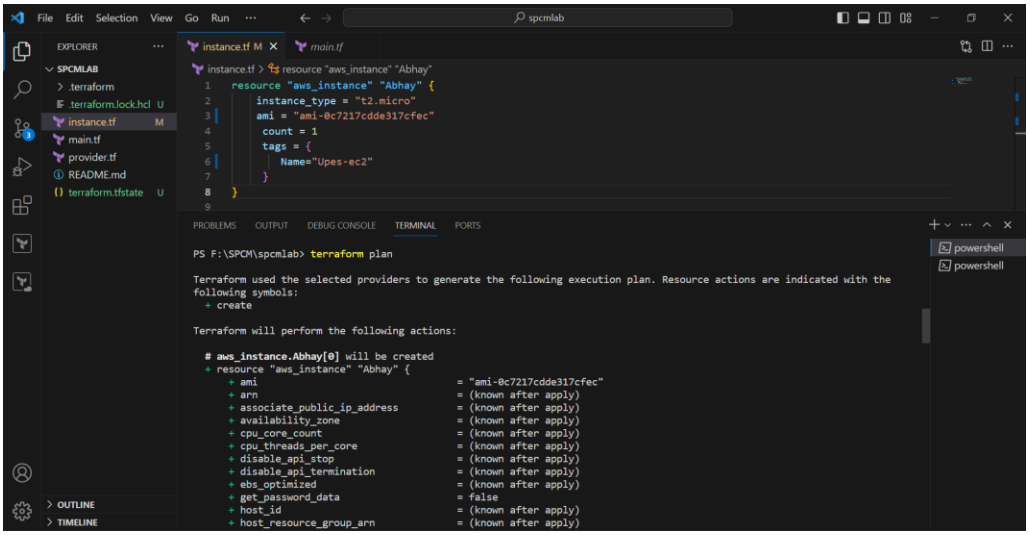
3: Initialize Terraform: Run the following command to initialize your Terraform working directory:

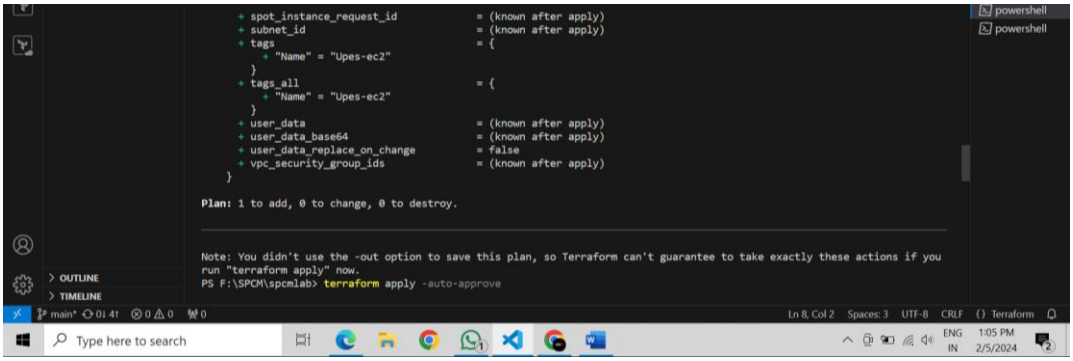
4: Create Terraform Configuration File for EC2 instance (instance.tf): Create a file named instnace.tf with the following content:



Step 5: Review Plan: Run the following command to see what Terraform will do:

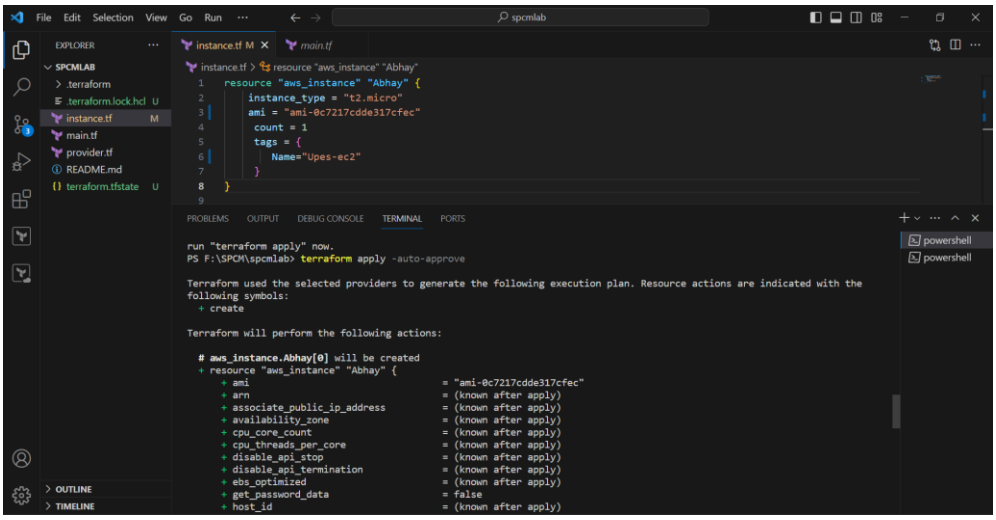
Review the plan to ensure it aligns with your expectations.

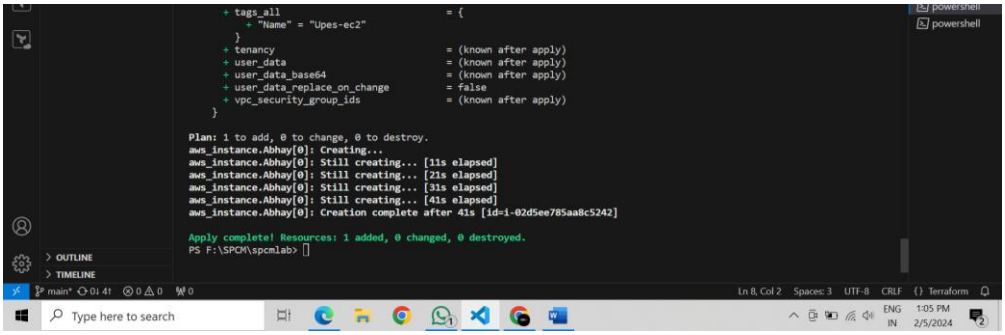




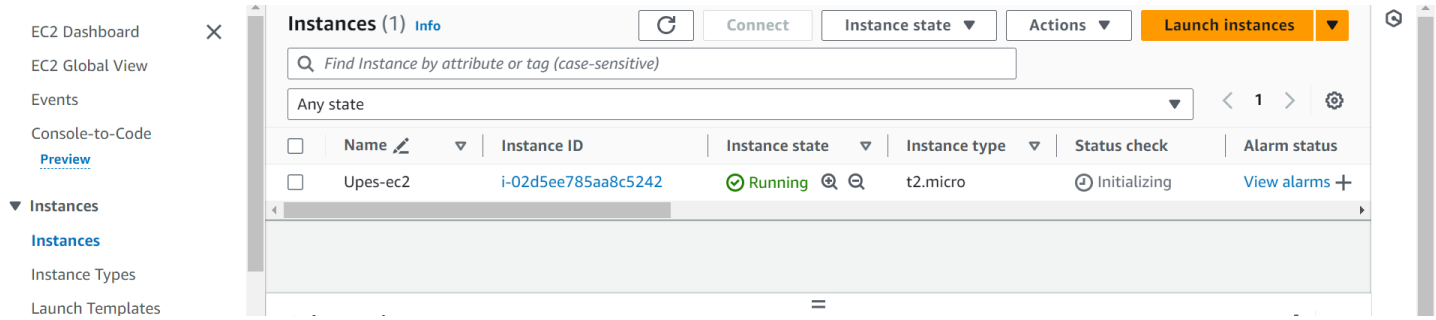
Step 6: Apply Changes: Apply the changes to create the AWS resources:

Type yes when prompted.

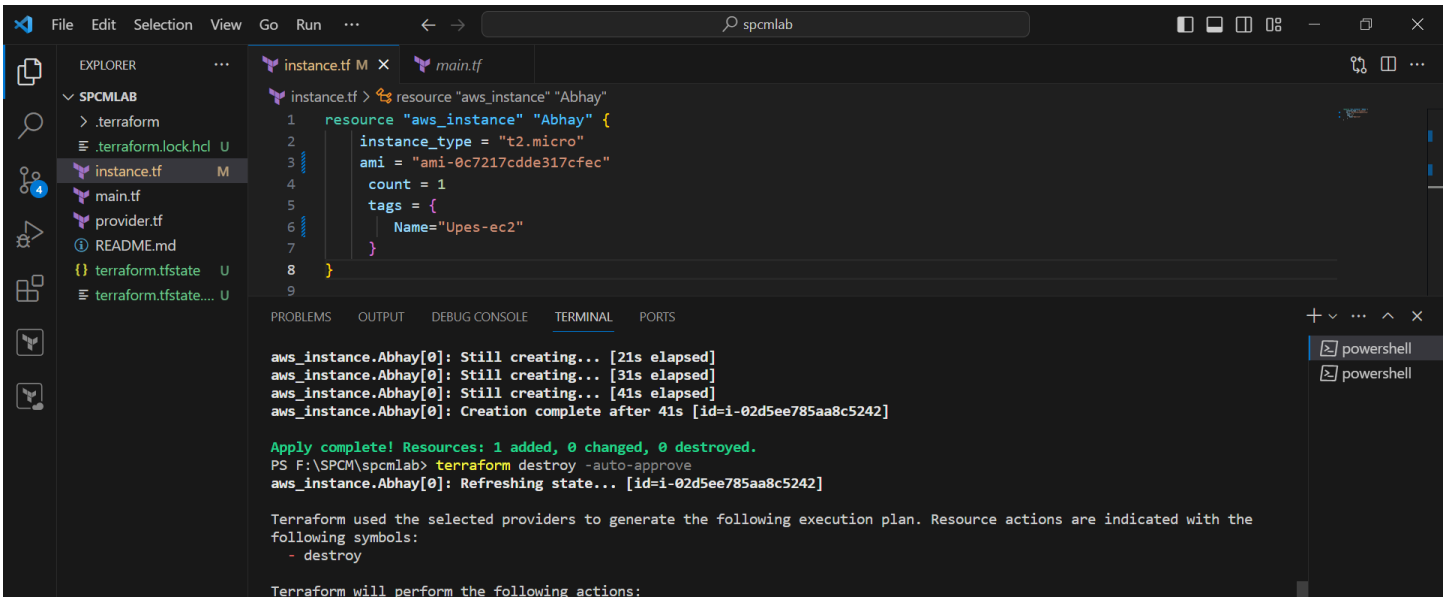


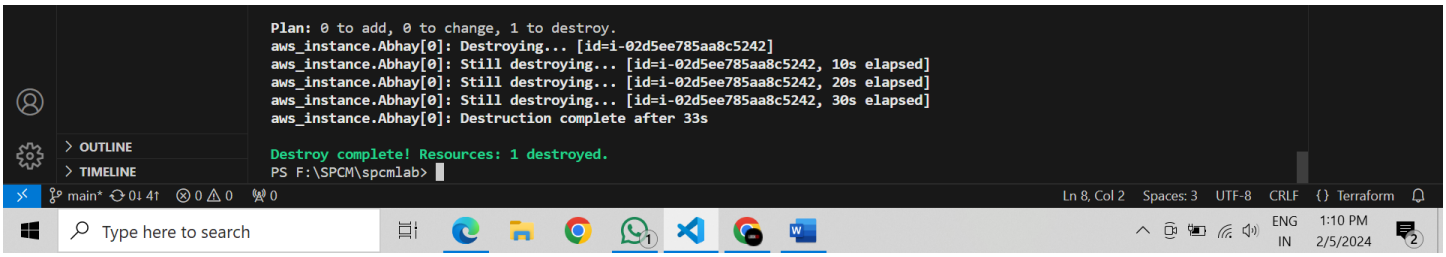


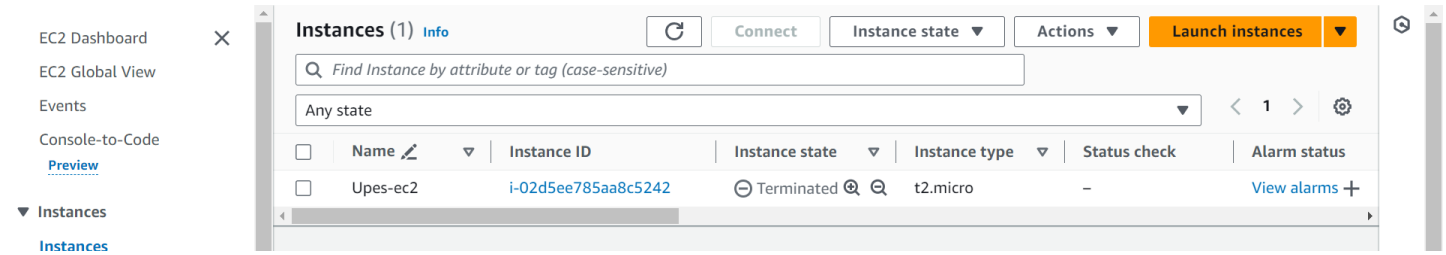
Step 7: Verify Resources: After the terraform apply command completes, log in to your AWS Management Console and navigate to the EC2 dashboard. Verify that the EC2 instance has been created.



Step 8: Cleanup Resources: When you are done experimenting, run the following command to destroy the created resources:


File Edit Selection View Go Run
spcmlab
EXPLORER
SPCMLAB
.terraform
.terraform.lock.hcl U
instance.tf M
main.tf
provider.tf
README.md
terraform.tfstate U
terraform.tfstate.... U
instance.tf M
main.tf
instance.tf > resource "aws_instance" "Abhay"
resource "aws_instance" "Abhay" {
instance_type = "t2.micro"
ami = "ami-0c7217cdde317cfec"
count = 1
tags = {
Name="Upes-ec2"
}
}
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
powershell
aws_instance.Abhay[0]: Still creating... [21s elapsed]
aws_instance.Abhay[0]: Still creating... [31s elapsed]
aws_instance.Abhay[0]: Still creating... [41s elapsed]
aws_instance.Abhay[0]: Creation complete after 41s [id=i-02d5ee785aa8c5242]
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
PS F:\SPCM\spcmlab> terraform destroy -auto-approve
aws_instance.Abhay[0]: Refreshing state... [id=i-02d5ee785aa8c5242]
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:


Plan: 0 to add, 0 to change, 1 to destroy.
aws_instance.Abhay[0]: Destroying... [id=i-02d5ee785aa8c5242]
aws_instance.Abhay[0]: Still destroying... [id=i-02d5ee785aa8c5242, 10s elapsed]
aws_instance.Abhay[0]: Still destroying... [id=i-02d5ee785aa8c5242, 20s elapsed]
aws_instance.Abhay[0]: Still destroying... [id=i-02d5ee785aa8c5242, 30s elapsed]
aws_instance.Abhay[0]: Destruction complete after 33s
Destroy complete! Resources: 1 destroyed.
PS F:\SPCM\spcmlab>
OUTLINE
TIMELINE
Ln 8, Col 2 Spaces: 3 UTF-8 CRLF {} Terraform
Type here to search
ENG IN
1:10 PM 2/5/2024


EC2 Dashboard
EC2 Global View
Events
Console-to-Code
Preview
Instances
Instances
Instances (1) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
Any state
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Upes-ec2
i-02d5ee785aa8c5242
Terminated
t2.micro
–
View alarms

Lab Exercise 4– Terraform Variables

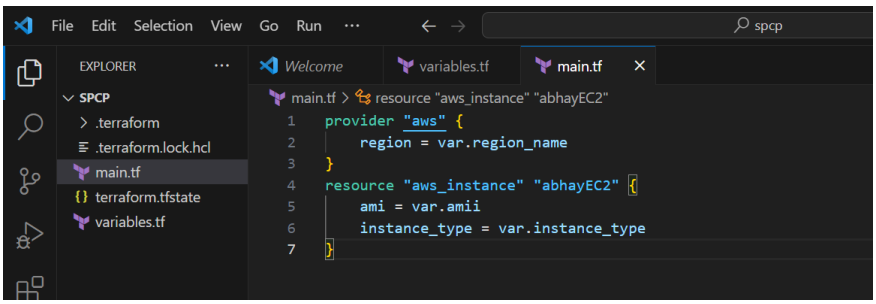
Objective: Learn how to define and use variables in Terraform configuration.

Steps: 1. Create a Terraform Directory:

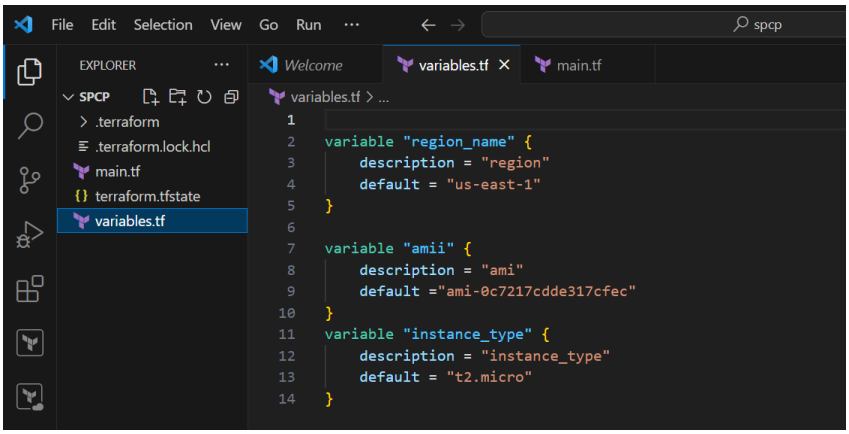
• Create a new directory for your Terraform project.

2. Create a Terraform Configuration File:

• Create a file named main.tf within your project directory. # main.tf

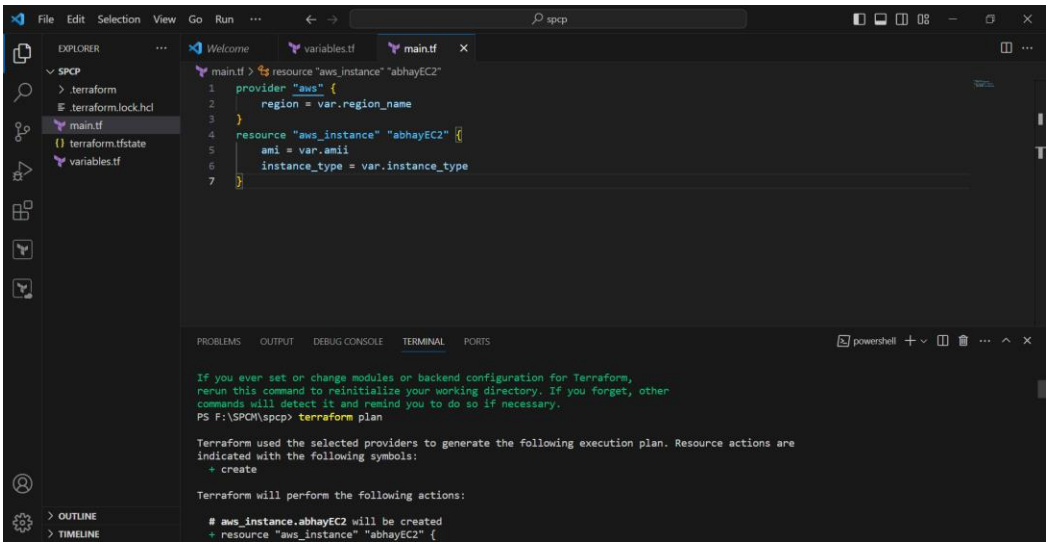


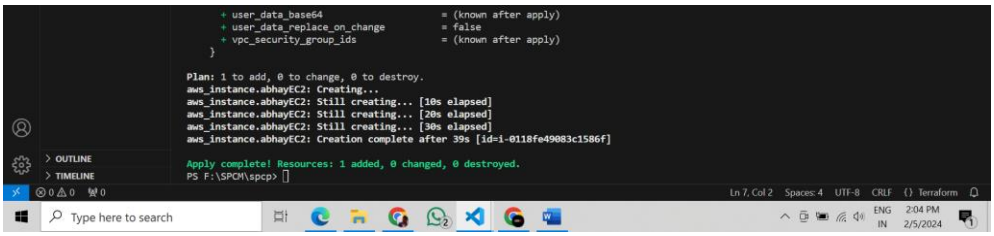
3. Define Variables: • Open a new file named variables.tf. Define variables for region, ami, and instance_type. # variables.tf

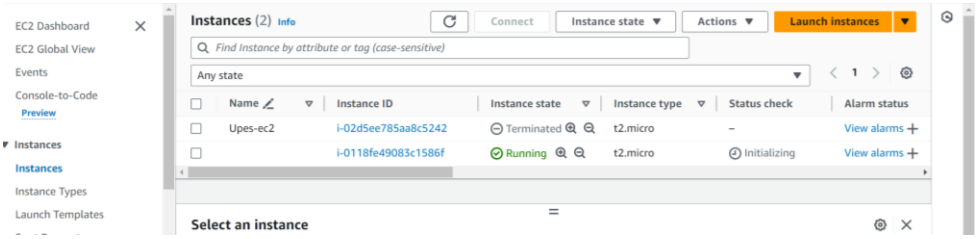


4. Use Variables in main.tf: • Modify main.tf to use the variables. # main.tf

5. Initialize and Apply: • Run the following Terraform commands to initialize and apply the configuration.

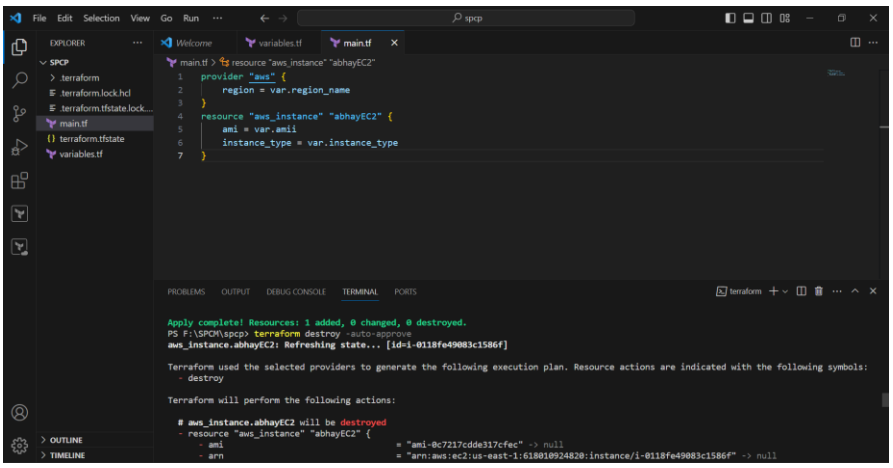


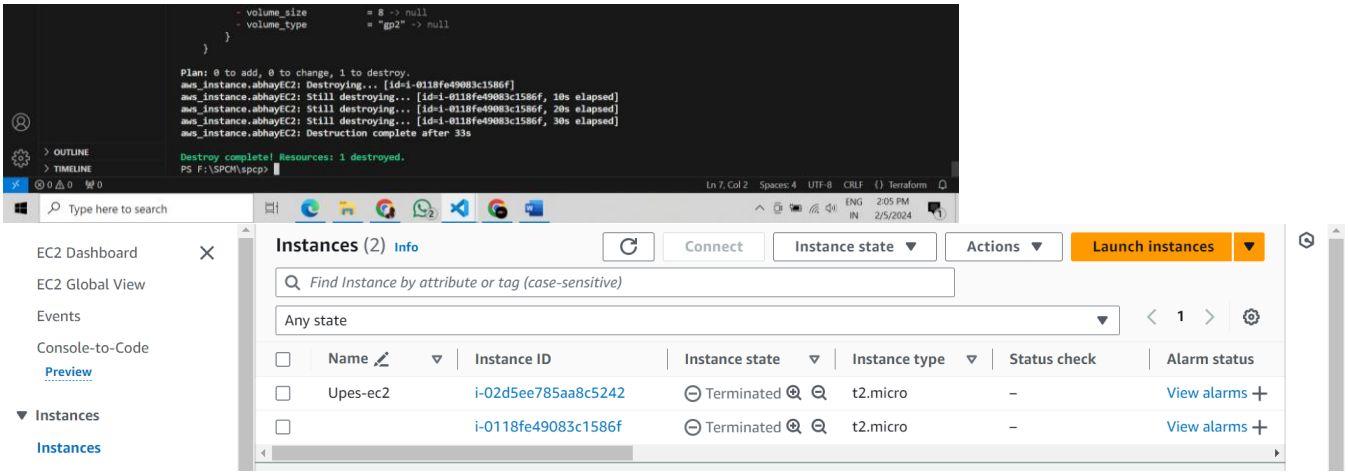




Observe how the region changes based on the variable override.

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

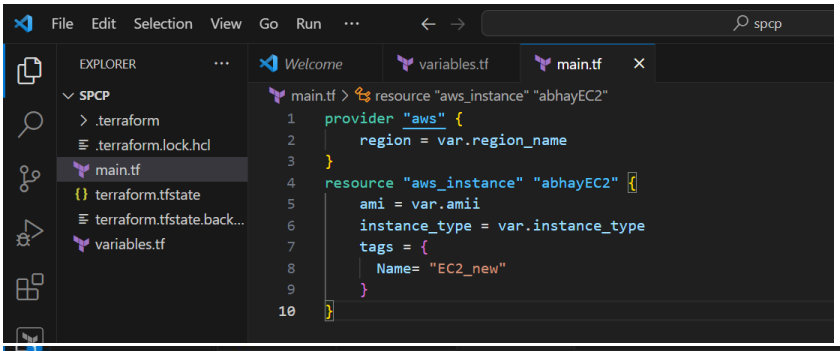


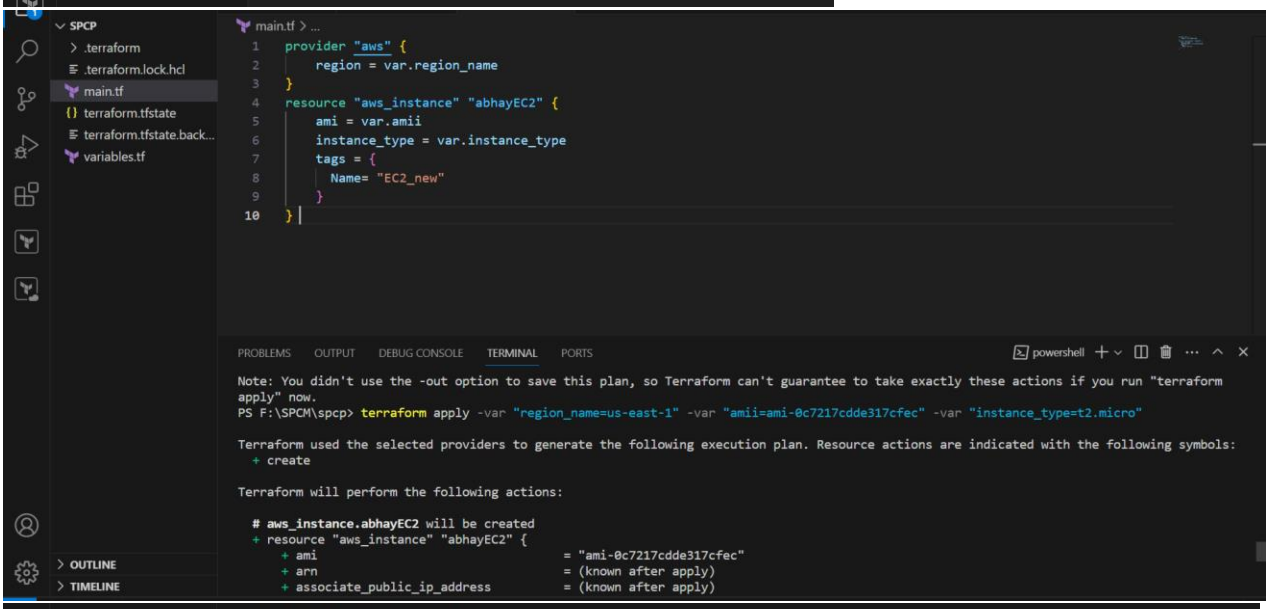


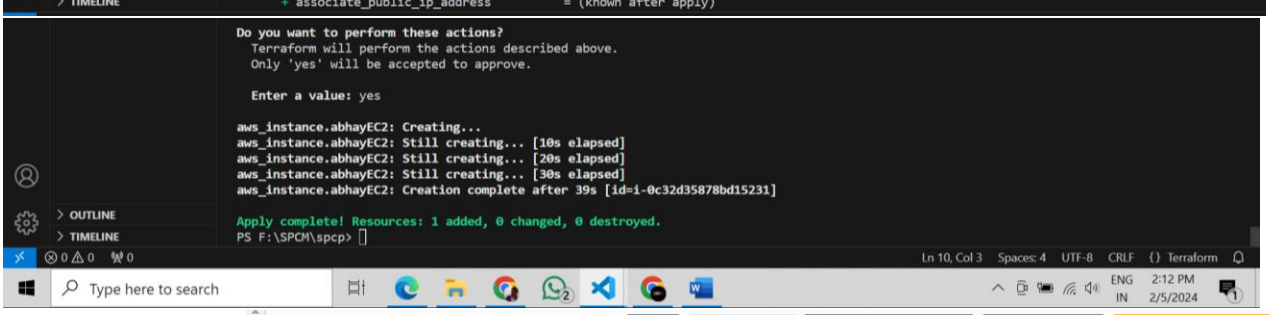
Lab Exercise 5– Terraform Variables with Command Line Arguments

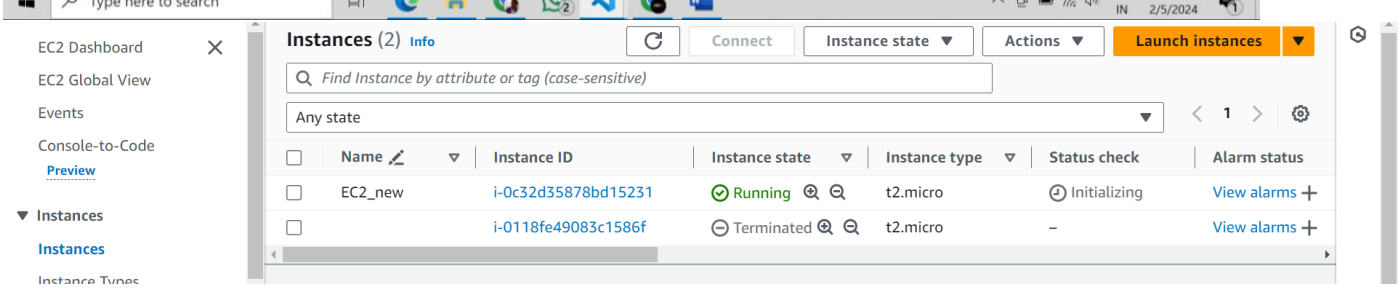
. Use Command Line Arguments: • Open a terminal and navigate to your Terraform project directory. • Run the terraform init command:

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









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

File Edit Selection View Go Run ··· spcp

EXPLORER

SPCP
- .terraform
- .terraform.lock.hcl
- .terraform.tfstate.lock....
- main.tf
- terraform.tfstate
- terraform.tfstate.back...
- variables.tf

Welcome | variables.tf | main.tf

main.tf > ...

```tf
provider "aws" {
    region = var.region_name
}
resource "aws_instance" "abhayEC2" {
    ami = var.amii
    instance_type = var.instance_type
    tags = {
      Name= "EC2_new"
    }
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
aws_instance.abhayEC2: Still creating... [20s elapsed]
aws_instance.abhayEC2: Still creating... [30s elapsed]
aws_instance.abhayEC2: Creation complete after 39s [id=i-0c32d35878bd15231]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
PS F:\SPCM\spcp> terraform destroy -auto-approve
aws_instance.abhayEC2: Refreshing state... [id=i-0c32d35878bd15231]

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.abhayEC2 will be destroyed
```

OUTLINE
TIMELINE

```
          - volume_size           = 8 -> null
          - volume_type           = "gp2" -> null
        }
    }

Plan: 0 to add, 0 to change, 1 to destroy.
aws_instance.abhayEC2: Destroying... [id=i-0c32d35878bd15231]
aws_instance.abhayEC2: Still destroying... [id=i-0c32d35878bd15231, 10s elapsed]
aws_instance.abhayEC2: Still destroying... [id=i-0c32d35878bd15231, 20s elapsed]
aws_instance.abhayEC2: Still destroying... [id=i-0c32d35878bd15231, 30s elapsed]
aws_instance.abhayEC2: Destruction complete after 34s

Destroy complete! Resources: 1 destroyed.
PS F:\SPCM\spcp>
```

OUTLINE
TIMELINE

⊗ 0 ⚠ 0 0 | Ln 10, Col 3 | Spaces: 4 | UTF-8 | CRLF | {} Terraform

Type here to search | ENG IN | 2:14 PM 2/5/2024

EC2 Dashboard
EC2 Global View
Events
Console-to-Code
Preview
Instances
Instances

Instances (2) Info

Connect | Instance state | Actions | Launch instances

Find Instance by attribute or tag (case-sensitive)

Any state

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
EC2_new	i-0c32d35878bd15231	Terminated	t2.micro	–	View alarms
	i-0118fe49083c1586f	Terminated	t2.micro	–	View alarms