

School of Computer Science
UNIVERSITY OF PETROLEUM AND ENERGY STUDIES
DEHRADUN, UTTARAKHAND



**System Monitoring And Configuration
Management**

Lab File
(2024-2025)

for
6th Semester

Submitted To:
Dr. Hitesh Kumar Sharma

Submitted By:
Parth Paul
B. Tech. CSE DevOps
- [6th Semester]
Sap id- 500091852
Batch 1
R2142210588

LAB EXERCISE 1

Aim: Install Terraform on Windows

Download Terraform, add it to path and verify install.

Windows

Binary download

386

Version: 1.7.2

Download 

AMD64

Version: 1.7.2

Download 

Linux

Package manager

Ubuntu/Debian










CentOS/RHEL

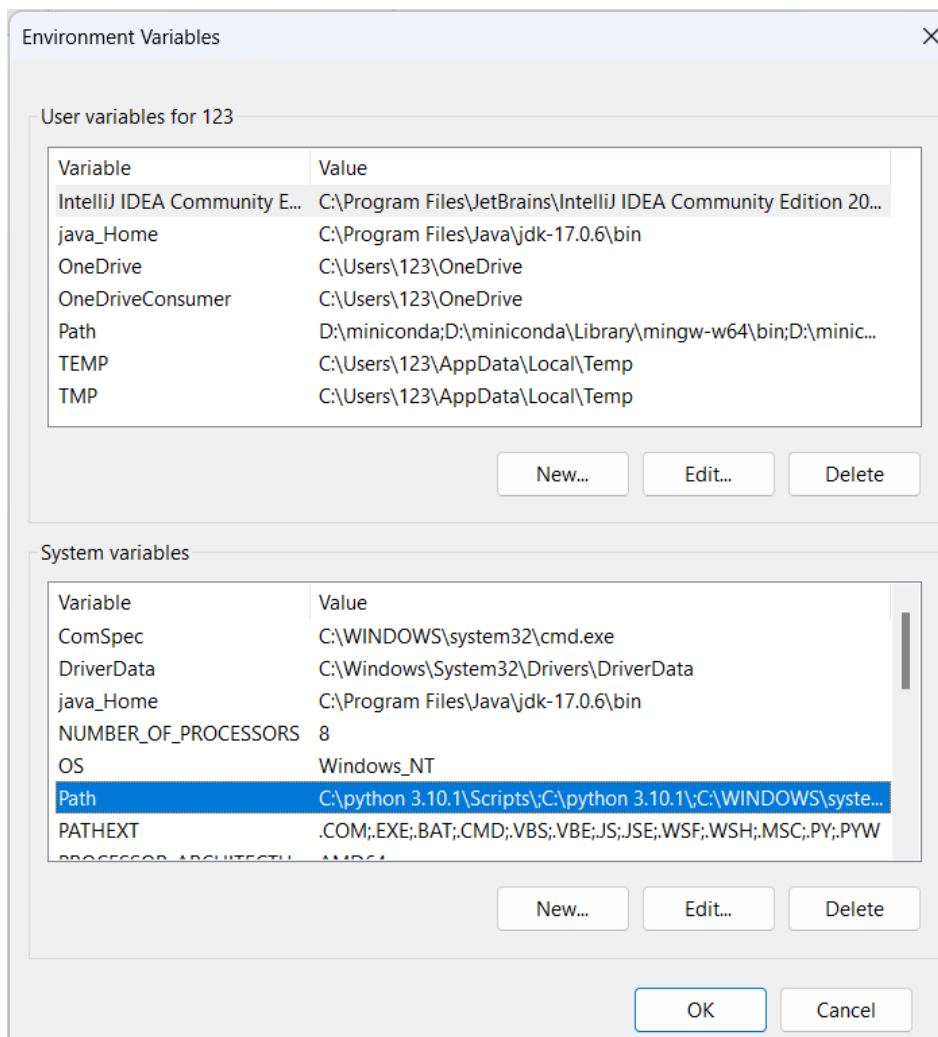
Fedora

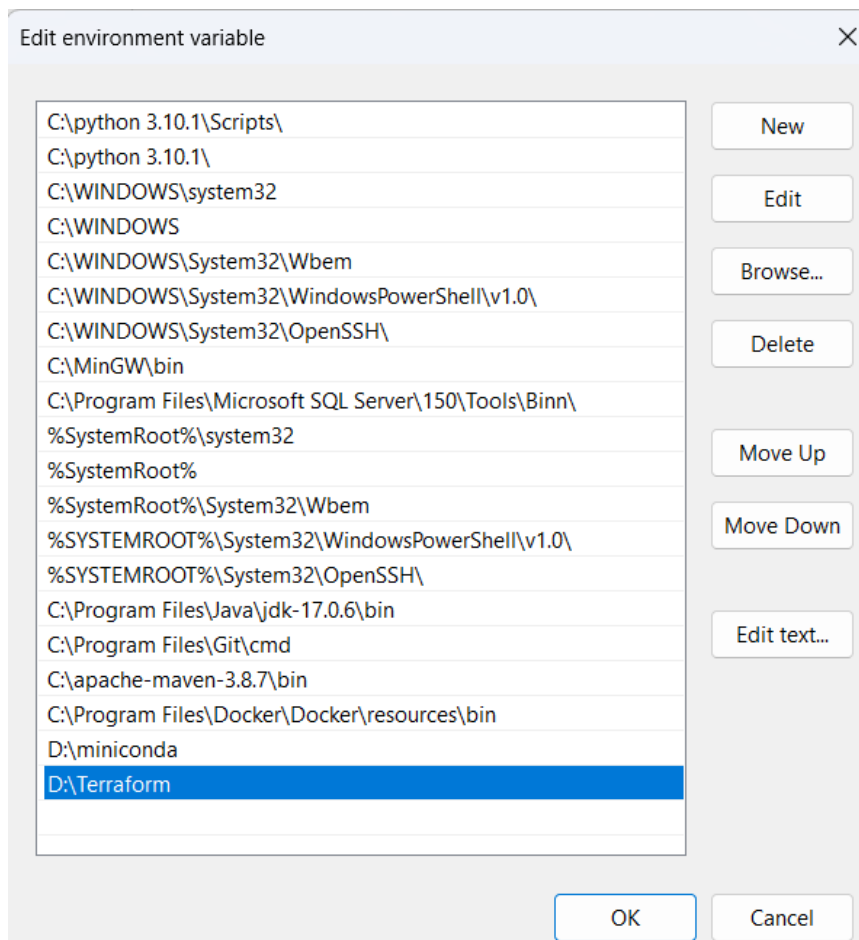
Amazon Linux

Homebrew

```
$ wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
$ echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.hashicorp.com/ubuntu/ terraform main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
$ sudo apt update && sudo apt install terraform
```

Downloads > terraform_1.6.6_windows_amd64				
       Sort  View 				
	Name	Date modified	Type	Size
▼ Last month	terraform	16-01-2024 09:09	Application	80,907 KB





On cmd run this command for the successful installation.

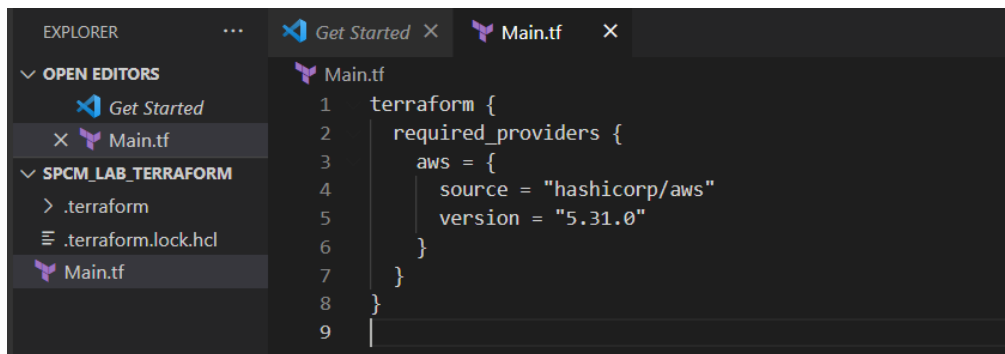
```
C:\Users\parth>terraform -version
Terraform v1.6.6
on windows_amd64

Your version of Terraform is out of date! The latest version
is 1.7.2. You can update by downloading from https://www.terraform.io/downloads.html
```

LAB EXERCISE 2

Aim: Terraform AWS Provider and IAM User Setting

Step 1: Create Terraform Configuration File (main.tf)



The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows a project named 'SPCM_LAB_TERRAFORM' with files '.terraform', '.terraform.lock.hcl', and 'Main.tf'. The 'Main.tf' file is open in the editor, showing the following Terraform configuration:

```
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "5.31.0"
6     }
7   }
8 }
9 |
```

Step 3: Initialize Terraform:

```
parth@LAPTOP-EL5DNB1S MINGW64 ~/aws-terraform-demo
$ terraform init
```

Initializing the backend...

Initializing provider plugins...

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.

```
parth@LAPTOP-EL5DNB1S MINGW64 ~/aws-terraform-demo
$ |
```

LAB EXERCISE 3

Aim: Provisioning an EC2 Instance on AWS

Step 1: Create Terraform Configuration File (main.tf)

```
main.tf
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "5.31.0"
6     }
7   }
8 }
9 provider "aws" {
10   region      = "ap-south-1"
11   access_key   = "AKIA55IIA4K2GLUL2CMG"
12   secret_key   = "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"
13 }
```

Step 2: Create Terraform Configuration File for EC2 instance (instance.tf)

```
main.tf  instance.tf X
instance.tf
1 resource "aws_instance" "My-instance" {
2   instance_type = "t2.micro"
3   ami = "ami-0d63de463e6604d0a"
4   count = 1
5   tags = {
6     Name = "UPES-EC2-Instnace"
7   }
8 }
```

Step 3: Initialize Terraform:

```
parth@LAPTOP-EL5DNB1S MINGW64 ~/aws-terraform-demo
$ terraform init
```

Initializing the backend...

Initializing provider plugins...

- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.31.0

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.

Step 4: Apply Validate

```
parth@LAPTOP-EL5DNB1S MINGW64 ~/aws-terraform-demo
$ terraform validate
Success! The configuration is valid.
```

Step 5: Review Plan:

```
parth@LAPTOP-EL5DNB1S MINGW64 ~/aws-terraform-demo
$ 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[0] will be created
+ resource "aws_instance" "My-instance" {
  + ami                        = "ami-0d63de463e6604d0a"
  + 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)
  + private_ip                 = (known after apply)
  + public_dns                 = (known after apply)
  + public_ip                  = (known after apply)
  + secondary_private_ips      = (known after apply)
```

```

+ subnet_id          = (known after apply)
+ tags               = {
  + "Name" = "UPES-EC2-Instnace"
}
+ tags_all           = {
  + "Name" = "UPES-EC2-Instnace"
}
+ 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)
}

```

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

Step 6: Apply Changes

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

Do you want to perform these actions?

Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_instance.My-Instnace[0]: Creating...

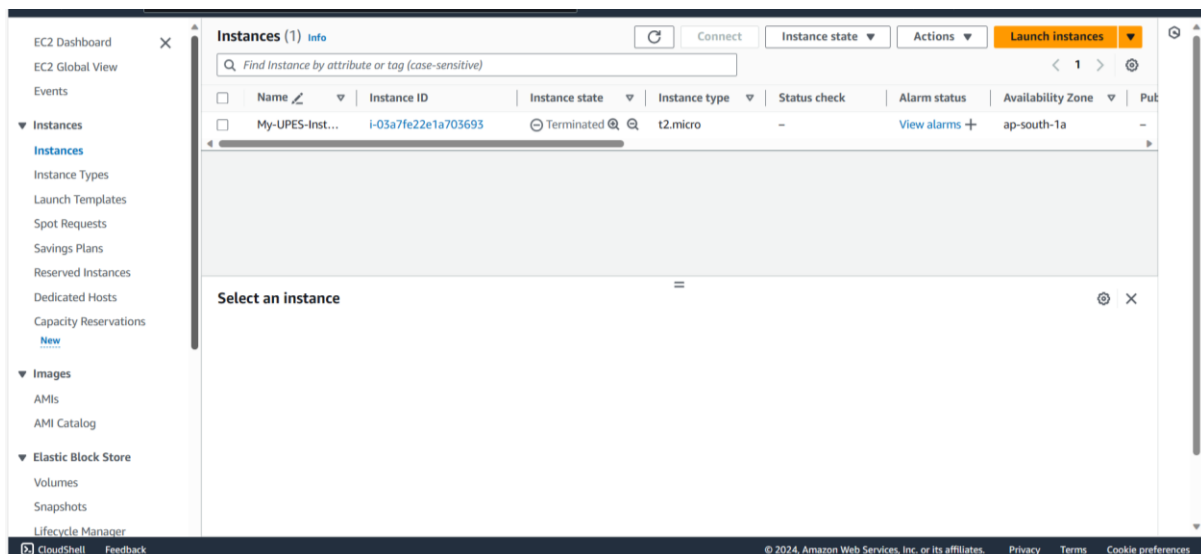
aws_instance.My-Instnace[0]: Still creating... [10s elapsed]

aws_instance.My-Instnace[0]: Still creating... [20s elapsed]

aws_instance.My-Instnace[0]: Still creating... [30s elapsed]

aws_instance.My-Instnace[0]: Creation complete after 40s [id=i-03a7fe22e1a703693]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.



Step 7: Cleanup Resources

Terraform will perform the following actions:

```
# aws_instance.My-Instnace[0] will be destroyed
- resource "aws_instance" "My-Instnace" {
  - ami                    = "ami-03f4878755434977f" -> null
  - arn                   = "arn:aws:ec2:ap-south-1:637423348062:instance/i-03a7fe22e1a703693" -> null
  - associate_public_ip_address = true -> null
  - availability_zone      = "ap-south-1a" -> 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-03a7fe22e1a703693" -> 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-0ebfddd302e87b053" -> null
  - private_dns            = "ip-172-31-35-24.ap-south-1.compute.internal" -> null
  - private_ip             = "172.31.35.24" -> null
  - public_dns             = "ec2-43-205-114-204.ap-south-1.compute.amazonaws.com" -> null
  - public_ip              = "43.205.114.204" -> null
  - secondary_private_ips  = [] -> null
  - security_groups        = [
    - "default",
  ] -> null
  - source_dest_check      = true -> null
  - subnet_id              = "subnet-0fb95688eaa188f7d" -> null
  - tags                   = {
    - "Name" = "My-UPES-Instnace"
  } -> null
  - tags_all               = {
    - "Name" = "My-UPES-Instnace"
  } -> null
  - tenancy                 = "default" -> null
  - user_data_replace_on_change = false -> null
  - vpc_security_group_ids  = [
    - "sg-0c6b5aae418c53ba2",
  ] -> null

  - capacity_reservation_specification {
    - capacity_reservation_preference = "open" -> null
  }

  - cpu_options {
    - core_count      = 1 -> null
    - threads_per_core = 1 -> null
  }

  - credit_specification {
    - cpu_credits = "standard" -> null
  }
}
```

```

- capacity_reservation_specification {
  - capacity_reservation_preference = "open" -> null
}

- cpu_options {
  - core_count      = 1 -> null
  - threads_per_core = 1 -> null
}

- credit_specification {
  - cpu_credits = "standard" -> null
}

- enclave_options {
  - enabled = false -> null
}

- maintenance_options {
  - auto_recovery = "default" -> null
}

- metadata_options {
  - http_endpoint      = "enabled" -> null
  - http_protocol_ipv6 = "disabled" -> null
  - http_put_response_hop_limit = 1 -> null
  - http_tokens        = "optional" -> null
  - instance_metadata_tags = "disabled" -> null
}

- private_dns_name_options {
  - enable_resource_name_dns_a_record = false -> null
  - enable_resource_name_dns_aaaa_record = false -> null
  - hostname_type                     = "ip-name" -> null
}

- root_block_device {
  - delete_on_termination = true -> null
  - device_name           = "/dev/sda1" -> null
  - encrypted             = false -> null
  - iops                  = 100 -> null
  - tags                  = {} -> null
  - throughput            = 0 -> null
  - volume_id             = "vol-082d5578223000e93" -> null
  - volume_size           = 8 -> null
  - volume_type           = "gp2" -> null
}
}

```

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

Do you really want to destroy all resources?

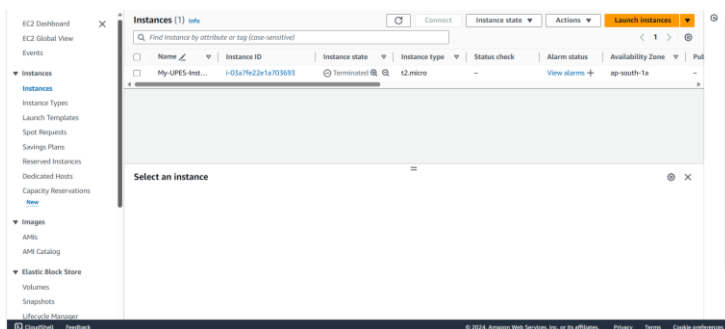
Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

```

aws_instance.My-Instnace[0]: Destroying... [id=i-03a7fe22e1a703693]
aws_instance.My-Instnace[0]: Still destroying... [id=i-03a7fe22e1a703693, 10s elapsed]
aws_instance.My-Instnace[0]: Still destroying... [id=i-03a7fe22e1a703693, 20s elapsed]
aws_instance.My-Instnace[0]: Still destroying... [id=i-03a7fe22e1a703693, 30s elapsed]
aws_instance.My-Instnace[0]: Destruction complete after 31s

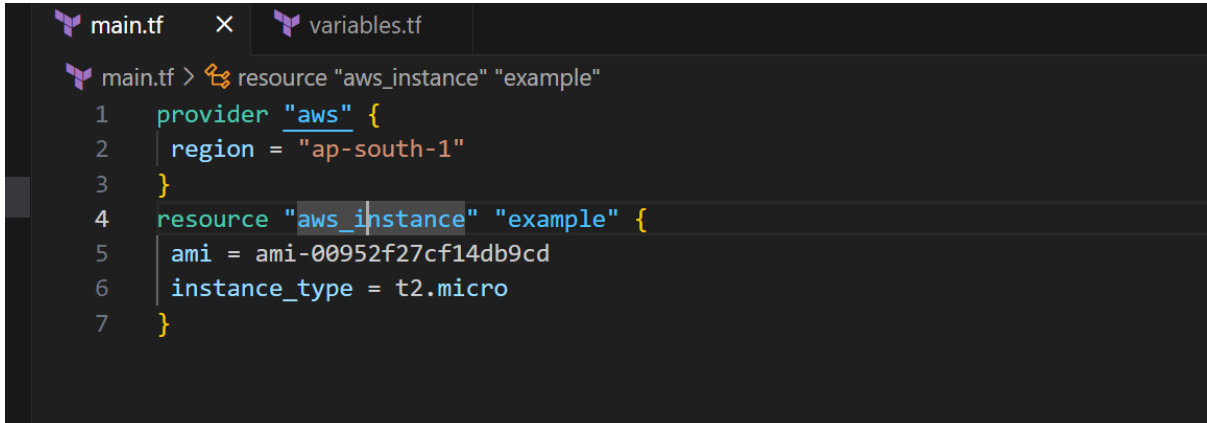
```



LAB EXERCISE 4

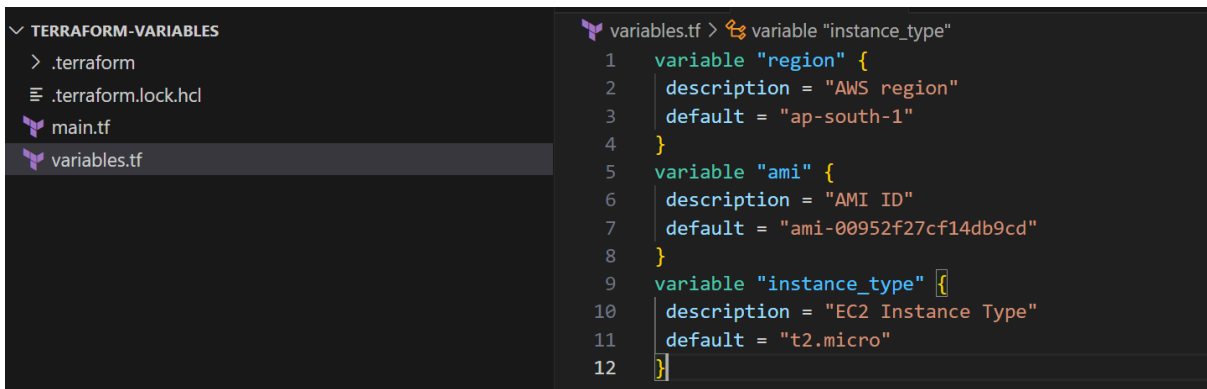
Aim: Terraform Variables

Step 1: Create a Terraform Configuration File:



```
main.tf > resource "aws_instance" "example"
1  provider "aws" {
2    region = "ap-south-1"
3  }
4  resource "aws_instance" "example" {
5    ami = ami-00952f27cf14db9cd
6    instance_type = t2.micro
7  }
```

Step 2: Define Variables:



```
variables.tf > variable "instance_type"
1  variable "region" {
2    description = "AWS region"
3    default = "ap-south-1"
4  }
5  variable "ami" {
6    description = "AMI ID"
7    default = "ami-00952f27cf14db9cd"
8  }
9  variable "instance_type" {
10   description = "EC2 Instance Type"
11   default = "t2.micro"
12 }
```

Step 3: Use Variables in instance.tf file:

```
Instance.tf > variable "ami" > default
1  resource "aws_instance" "My-Instnace-01" {
2      instance_type = var.instance_ty
3      ami = var.ami
4      count= var.instance_count
5      tags = {
6          Name = "UPES-EC2-Instnace"
7      }
8  }
9
10 resource "aws_instance" "My-Instnace-02" {
11     instance_type = var.instance_ty
12     ami = var.ami
13     count= var.instance_count
14
15     tags = {
16         Name = "UPES-EC2-Instnace"
17     }
18 }
19
20 resource "aws_instance" "My-Instnace-03" {
21     instance_type = var.instance_ty
22     ami = var.ami
23     count= var.instance_count
24
25     tags = {
26         Name = "UPES-EC2-Instnace"
27     }
28 }
29
```

Step 4: Now do terraform initialize:

```
Microsoft Windows [Version 10.0.22621.3007]
(c) Microsoft Corporation. All rights reserved.

F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform validate
Success! The configuration is valid.
```

Step 5: Now perform Terraform plan:

Acquiring state lock. This may take a few moments...

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-Instnace-01[0] will be created

```
+ resource "aws_instance" "My-Instnace-01" {
  + ami                    = "ami-03f4878755434977f"
  + 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)
  + 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" = "UPES-EC2-Instnace"
  }
  + tags_all              = {
    + "Name" = "UPES-EC2-Instnace"
  }
  + tenancy                = (known after apply)
  + user_data              = (known after apply)
```

```

    }
    + 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)
  }

# aws_instance.My-Instnace-02[0] will be created
+ resource "aws_instance" "My-Instnace-02" {
  + ami                        = "ami-03f4878755434977f"
  + 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)
  + 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" = "UPES-EC2-Instnace"
  }
  + tags_all                   = {
    + "Name" = "UPES-EC2-Instnace"
  }
  + tenancy                    = (known after apply)
  + user_data                  = (known after apply)
  + user_data_base64          = (known after apply)
  + user_data_replace_on_change = false

```

```

    + "Name" = "UPES-EC2-Instnace"
  }
+ tags_all = {
  + "Name" = "UPES-EC2-Instnace"
}
+ 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)
}

# aws_instance.My-Instnace-03[0] will be created
+ resource "aws_instance" "My-Instnace-03" {
  + ami = "ami-03f4878755434977f"
  + 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)
  + 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" = "UPES-EC2-Instnace"
  }
+ tags_all = {
  + "Name" = "UPES-EC2-Instnace"
}
}

```

```
17. Get 0348787554349774f ami-0348787554349774f
```

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-Instnace-01[0] will be created
+ resource "aws_instance" "My-Instnace-01" {
+   ami                     = "ami-0348787554349774f"
+   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)
+   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" = "UPES-EC2-Instnace"
+   }
}
```



```

+ tags_all = {
+   + "Name" = "UPES-EC2-Instnace"
+ }
+ 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)
}

# aws_instance.My-Instnace-02[0] will be created
+ resource "aws_instance" "My-Instnace-02" {
+   ami = "ami-03f4878755434977f"
+   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)
+   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" = "UPES-EC2-Instnace"
+   }
+   tags_all = {
+     + "Name" = "UPES-EC2-Instnace"
+   }
+   tenancy = (known after apply)
+   user_data = (known after apply)
}

```

```

+ vpc_security_group_ids      = (known after apply)
}

# aws_instance.My-Instnace-03[0] will be created
+ resource "aws_instance" "My-Instnace-03" {
+   ami                      = "ami-03f4878755434977f"
+   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)
+   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" = "UPES-EC2-Instnace"
+   }
+   tags_all                 = {
+     "Name" = "UPES-EC2-Instnace"
+   }
+   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)
}

```

Plan: 3 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

```

aws_instance.My-Instnace-03[0]: Creating...
aws_instance.My-Instnace-01[0]: Creating...
aws_instance.My-Instnace-02[0]: Creating...
aws_instance.My-Instnace-02[0]: Still creating... [10s elapsed]
aws_instance.My-Instnace-01[0]: Still creating... [10s elapsed]
aws_instance.My-Instnace-03[0]: Still creating... [10s elapsed]
aws_instance.My-Instnace-02[0]: Still creating... [20s elapsed]
aws_instance.My-Instnace-03[0]: Still creating... [20s elapsed]
aws_instance.My-Instnace-01[0]: Still creating... [20s elapsed]
aws_instance.My-Instnace-02[0]: Creation complete after 26s [id=i-0a2c04548c6185370]
aws_instance.My-Instnace-01[0]: Creation complete after 27s [id=i-0159152f0199b756a]
aws_instance.My-Instnace-03[0]: Still creating... [30s elapsed]
aws_instance.My-Instnace-03[0]: Creation complete after 38s [id=i-01c6be481182ff8cd]

```

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.

EC2 Dashboard

EC2 Global View

Events

▼ Instances

Instances

Instance Types

Launch Templates

Spot Requests

Instances (3) info

Refresh

Connect

Instance state ▼

Actions ▼

Launch instances ▼

Find Instance by attribute or tag (case-sensitive)

Any state ▼

< 1 > ⚙

<input type="checkbox"/>	Name ↗	Instance ID	Instance state ▼	Instance type ▼	Status check	Alarm status	Availability Zone ▼	Public I
<input type="checkbox"/>	UPES-EC2-Inst...	i-0a2c04548c6185370	Running	t2.micro	⌚ Initializing	View alarms +	ap-south-1a	ec2-13-
<input type="checkbox"/>	UPES-EC2-Inst...	i-0159152f0199b756a	Running	t2.micro	⌚ Initializing	View alarms +	ap-south-1a	ec2-65-
<input type="checkbox"/>	UPES-EC2-Inst...	i-01c6be481182ff8cd	Running	t2.micro	⌚ Initializing	View alarms +	ap-south-1a	ec2-13-

Step 6: Now perform Terraform Destroy to clean up:

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform destroy
aws_instance.My-Instnace-03[0]: Refreshing state... [id=i-01c6be481182ff8cd]
aws_instance.My-Instnace-01[0]: Refreshing state... [id=i-0159152f0199b756a]
aws_instance.My-Instnace-02[0]: Refreshing state... [id=i-0a2c04548c6185370]

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-Instnace-01[0] will be destroyed
- resource "aws_instance" "My-Instnace-01" {
  - ami                      = "ami-0344878755434977f" -> null
  - arn                     = "arn:aws:ec2:ap-south-1:637423348062:instance/i-0159152f0199b756a" -> null
  - associate_public_ip_address = true -> null
  - availability_zone        = "ap-south-1a" -> 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-0159152f0199b756a" -> 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-09ddale69e703ba3f3" -> null
  - private_dns              = "ip-172-31-46-118.ap-south-1.compute.internal" -> null
  - private_ip               = "172.31.46.118" -> null
  - public_dns               = "ec2-65-0-92-98.ap-south-1.compute.amazonaws.com" -> null
  - public_ip                = "65.0.92.98" -> null
  - secondary_private_ips    = [] -> null
  - security_groups          = [
    - "default",
  ] -> null
  - source_dest_check        = true -> null
  - subnet_id                = "subnet-0fb95688eaa188f7d" -> null
  - tags                     = {
    - "Name" = "UPES-EC2-Instnace"
  } -> null
  - tags_all                 = {
    - "Name" = "UPES-EC2-Instnace"
  } -> null
  - tenancy                  = "default" -> null
  - user_data_replace_on_change = false -> null
  - vpc_security_group_ids   = [
    - "sg-0c6b5aae418c53ba2",
  ] -> null

- capacity_reservation_specification {
```

```

- capacity_reservation_specification {
  - capacity_reservation_preference = "open" -> null
}

- cpu_options {
  - core_count = 1 -> null
  - threads_per_core = 1 -> null
}

- credit_specification {
  - cpu_credits = "standard" -> null
}

- enclave_options {
  - enabled = false -> null
}

- maintenance_options {
  - auto_recovery = "default" -> null
}

- metadata_options {
  - http_endpoint = "enabled" -> null
  - http_protocol_ipv6 = "disabled" -> null
  - http_put_response_hop_limit = 1 -> null
  - http_tokens = "optional" -> null
  - instance_metadata_tags = "disabled" -> null
}

- private_dns_name_options {
  - enable_resource_name_dns_a_record = false -> null
  - enable_resource_name_dns_aaaa_record = false -> null
  - hostname_type = "ip-name" -> null
}

- root_block_device {
  - delete_on_termination = true -> null
  - device_name = "/dev/sda1" -> null
  - encrypted = false -> null
  - iops = 100 -> null
  - tags = {} -> null
  - throughput = 0 -> null
  - volume_id = "vol-04d6479744095f96c" -> null
  - volume_size = 8 -> null
  - volume_type = "gp2" -> null
}
}

# aws_instance.My-Instnace-02[0] will be destroyed
- resource "aws_instance" "My-Instnace-02" {
  - ami = "ami-03f4878755434977f" -> null
  - arn = "arn:aws:ec2:ap-south-1:637423348062:instance/i-0a2c04548c6185370" -> null
  - associate_public_ip_address = true -> null
  - availability_zone = "ap-south-1a" -> 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-0a2c04548c6185370" -> 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-09883271674178793" -> null
- private_dns = "ip-172-31-45-164.ap-south-1.compute.internal" -> null
- private_ip = "172.31.45.164" -> null
- public_dns = "ec2-13-234-204-23.ap-south-1.compute.amazonaws.com" -> null
- public_ip = "13.234.204.23" -> null
- secondary_private_ips = [] -> null
- security_groups = [
  - "default",
] -> null
- source_dest_check = true -> null
- subnet_id = "subnet-0fb95688eaa188f7d" -> null
- tags = {
  - "Name" = "UPES-EC2-Instnace"
} -> null
- tags_all = {
  - "Name" = "UPES-EC2-Instnace"
} -> null
- tenancy = "default" -> null
- user_data_replace_on_change = false -> null
- vpc_security_group_ids = [
  - "sg-0c6b5aae418c53ba2",
] -> null

- capacity_reservation_specification {
  - capacity_reservation_preference = "open" -> null
}

- cpu_options {
  - core_count = 1 -> null
  - threads_per_core = 1 -> null
}

- credit_specification {
  - cpu_credits = "standard" -> null
}

- enclave_options {
  - enabled = false -> null
}

- maintenance_options {
  - auto_recovery = "default" -> null
}

```

```

- metadata_options {
  - http_endpoint           = "enabled" -> null
  - http_protocol_ipv6      = "disabled" -> null
  - http_put_response_hop_limit = 1 -> null
  - http_tokens             = "optional" -> null
  - instance_metadata_tags   = "disabled" -> null
}

- private_dns_name_options {
  - enable_resource_name_dns_a_record = false -> null
  - enable_resource_name_dns_aaaa_record = false -> null
  - hostname_type                     = "ip-name" -> null
}

- root_block_device {
  - delete_on_termination = true -> null
  - device_name           = "/dev/sda1" -> null
  - encrypted             = false -> null
  - iops                  = 100 -> null
  - tags                  = {} -> null
  - throughput            = 0 -> null
  - volume_id             = "vol-0fd47691f9f057c4c" -> null
  - volume_size           = 8 -> null
  - volume_type           = "gp2" -> null
}
}

# aws_instance.My-Instnace-03[0] will be destroyed
- resource "aws_instance" "My-Instnace-03" {
  - ami                    = "ami-03f4878755434977f" -> null
  - arn                   = "arn:aws:ec2:ap-south-1:637423348062:instance/i-01c6be481182ff8cd" -> null
  - associate_public_ip_address = true -> null
  - availability_zone      = "ap-south-1a" -> 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-01c6be481182ff8cd" -> 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-04af9f7a4fa3a82f4" -> null
  - private_dns            = "ip-172-31-36-27.ap-south-1.compute.internal" -> null
  - private_ip             = "172.31.36.27" -> null
  - public_dns             = "ec2-13-127-181-85.ap-south-1.compute.amazonaws.com" -> null
  - public_ip              = "13.127.181.85" -> null
  - secondary_private_ips  = [] -> null
  - security_groups        = [
    - "default",

```

Plan: 0 to add, 0 to change, 3 to destroy.

Do you really want to destroy all resources?

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-Instnace-03[0]: Destroying... [id=i-01c6be481182ff8cd]
aws_instance.My-Instnace-01[0]: Destroying... [id=i-0159152f0199b756a]
aws_instance.My-Instnace-02[0]: Destroying... [id=i-0a2c04548c6185370]
aws_instance.My-Instnace-03[0]: Still destroying... [id=i-01c6be481182ff8cd, 10s elapsed]
aws_instance.My-Instnace-01[0]: Still destroying... [id=i-0159152f0199b756a, 10s elapsed]
aws_instance.My-Instnace-02[0]: Still destroying... [id=i-0a2c04548c6185370, 10s elapsed]
aws_instance.My-Instnace-02[0]: Still destroying... [id=i-0a2c04548c6185370, 20s elapsed]
aws_instance.My-Instnace-03[0]: Still destroying... [id=i-01c6be481182ff8cd, 20s elapsed]
aws_instance.My-Instnace-01[0]: Still destroying... [id=i-0159152f0199b756a, 20s elapsed]
aws_instance.My-Instnace-03[0]: Still destroying... [id=i-01c6be481182ff8cd, 30s elapsed]
aws_instance.My-Instnace-02[0]: Still destroying... [id=i-0a2c04548c6185370, 30s elapsed]
aws_instance.My-Instnace-01[0]: Still destroying... [id=i-0159152f0199b756a, 30s elapsed]
aws_instance.My-Instnace-01[0]: Destruction complete after 31s
aws_instance.My-Instnace-02[0]: Still destroying... [id=i-0a2c04548c6185370, 40s elapsed]
aws_instance.My-Instnace-03[0]: Still destroying... [id=i-01c6be481182ff8cd, 40s elapsed]
aws_instance.My-Instnace-02[0]: Destruction complete after 41s
aws_instance.My-Instnace-03[0]: Destruction complete after 41s

```

Destroy complete! Resources: 3 destroyed.

EC2 Dashboard

EC2 Global View

Events

▼ Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Instances (3) info

Connect

Instance state

Actions

Launch instances

Find Instance by attribute or tag (case-sensitive)

Any state

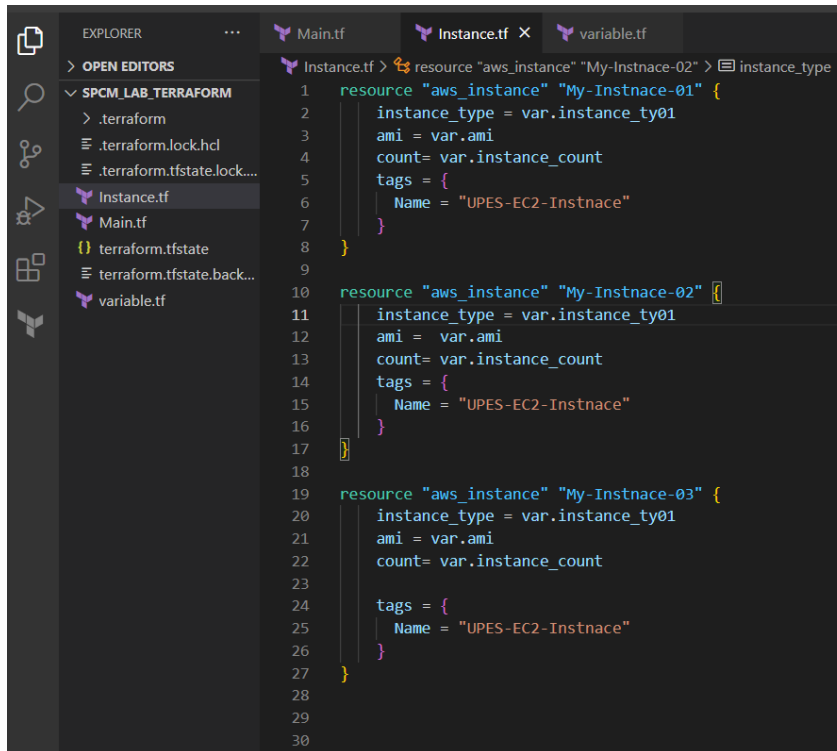
< 1 >

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	UPES-EC2-Inst...	i-0a2c04548c6185370	Terminated	t2.micro	-	View alarms +	ap-south-1a
<input type="checkbox"/>	UPES-EC2-Inst...	i-0159152f0199b756a	Terminated	t2.micro	-	View alarms +	ap-south-1a
<input type="checkbox"/>	UPES-EC2-Inst...	i-01c6be481182ff8cd	Terminated	t2.micro	-	View alarms +	ap-south-1a

LAB EXERCISE 5

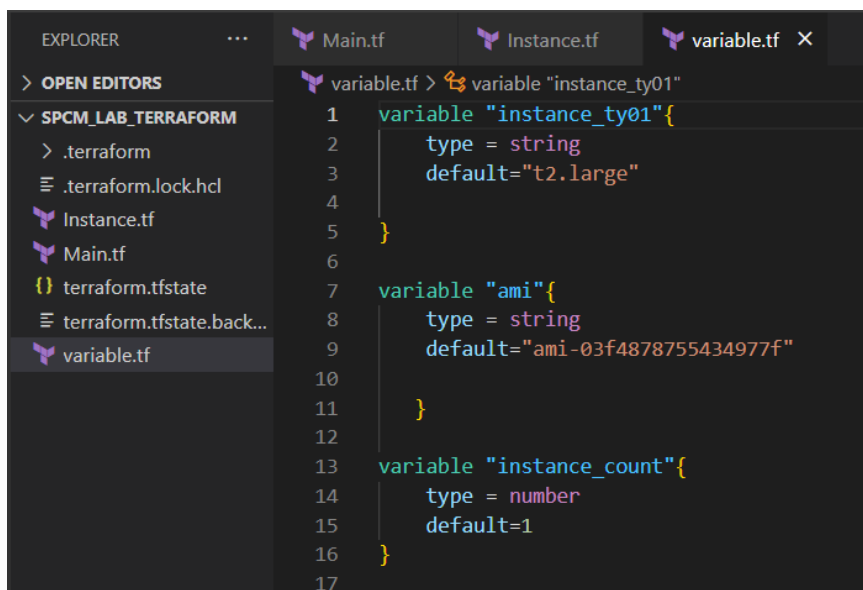
Aim: Terraform Variables with Command Line Arguments

Step 1: Create a instance.tf file:



```
1 resource "aws_instance" "My-Instnace-01" {
2     instance_type = var.instance_ty01
3     ami = var.ami
4     count= var.instance_count
5     tags = {
6         Name = "UPES-EC2-Instnace"
7     }
8 }
9
10 resource "aws_instance" "My-Instnace-02" {
11     instance_type = var.instance_ty01
12     ami = var.ami
13     count= var.instance_count
14     tags = {
15         Name = "UPES-EC2-Instnace"
16     }
17 }
18
19 resource "aws_instance" "My-Instnace-03" {
20     instance_type = var.instance_ty01
21     ami = var.ami
22     count= var.instance_count
23
24     tags = {
25         Name = "UPES-EC2-Instnace"
26     }
27 }
28
29
30
```

Step 2: Create a variable.tf file



```
1 variable "instance_ty01" {
2     type = string
3     default="t2.large"
4 }
5
6
7 variable "ami" {
8     type = string
9     default="ami-03f4878755434977f"
10 }
11
12
13 variable "instance_count" {
14     type = number
15     default=1
16 }
17
```

Step 3: Perform Terraform Validate And Apply

```
F:\SEM 6\SPCM LAB\SPCM LAB_TERRAFORM>terraform validate
Success! The configuration is valid.
```

```
F:\SEM 6\SPCM LAB\SPCM LAB_TERRAFORM>terraform apply
Acquiring state lock. This may take a few moments...
```

```
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-01[0] will be created
+ resource "aws_instance" "My-Instance-01" {
+   ami               = "ami-03f4878755434977f"
+   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.large"
+   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)
```

```

+ source_dest_check           = true
+ spot_instance_request_id    = (known after apply)
+ subnet_id                   = (known after apply)
+ tags                         = {
+   + "Name" = "UPES-EC2-Instnace"
+ }
+ tags_all                     = {
+   + "Name" = "UPES-EC2-Instnace"
+ }
+ 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)
}

# aws_instance.My-Instnace-02[0] will be created
+ resource "aws_instance" "My-Instnace-02" {
+   ami                      = "ami-03f4878755434977f"
+   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.large"
+   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)

```



```

+ 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" = "UPES-EC2-Instnace"
}
+ tags_all = {
  + "Name" = "UPES-EC2-Instnace"
}
+ 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)
}

# aws_instance.My-Instnace-03[0] will be created
+ resource "aws_instance" "My-Instnace-03" {
  + ami = "ami-03f4878755434977f"
  + 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.large"
  + ipv6_address_count = (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" = "UPES-EC2-Instnace"
}
+ tags_all = {
  + "Name" = "UPES-EC2-Instnace"
}
+ 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)
}

```

Plan: 3 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

```

aws_instance.My-Instnace-03[0]: Creating...
aws_instance.My-Instnace-01[0]: Creating...
aws_instance.My-Instnace-02[0]: Creating...
aws_instance.My-Instnace-02[0]: Still creating... [10s elapsed]
aws_instance.My-Instnace-03[0]: Still creating... [10s elapsed]
aws_instance.My-Instnace-01[0]: Still creating... [10s elapsed]
aws_instance.My-Instnace-03[0]: Still creating... [20s elapsed]
aws_instance.My-Instnace-01[0]: Still creating... [20s elapsed]
aws_instance.My-Instnace-02[0]: Still creating... [20s elapsed]
aws_instance.My-Instnace-01[0]: Creation complete after 24s [id=i-0edc01737ec2fe49a]
aws_instance.My-Instnace-03[0]: Creation complete after 24s [id=i-019432b41727b66a0]
aws_instance.My-Instnace-02[0]: Creation complete after 24s [id=i-0513ee647c371165f]

```

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.

EC2 Dashboard	Instances (6) info	Connect	Instance state	Actions	Launch instances
EC2 Global View	Find Instance by attribute or tag (case-sensitive)	Any state			
▼ Instances					
Instances					
Instance Types					
Launch Templates					
Spot Requests					
Savings Plans					
Reserved Instances					

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	UPES-EC2-Inst...	I-0a2c04548c6185370	Terminated	t2.micro	-	View alarms +	ap-south-1a
<input type="checkbox"/>	UPES-EC2-Inst...	I-0159152f0199b756a	Terminated	t2.micro	-	View alarms +	ap-south-1a
<input type="checkbox"/>	UPES-EC2-Inst...	I-01c6be481182ff8cd	Terminated	t2.micro	-	View alarms +	ap-south-1a
<input type="checkbox"/>	UPES-EC2-Inst...	I-0513ee647c371165f	Running	t2.large	Initializing	View alarms +	ap-south-1b
<input type="checkbox"/>	UPES-EC2-Inst...	I-019432b41727b66a0	Running	t2.large	Initializing	View alarms +	ap-south-1b
<input type="checkbox"/>	UPES-EC2-Inst...	I-0edc01737ec2fe49a	Running	t2.large	2/2 checks passed	View alarms +	ap-south-1b

Step 4: Perform Terraform Destroy:

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform destroy
aws_instance.My-Instance-02[0]: Refreshing state... [id=i-0513ee647c371165f]
aws_instance.My-Instance-03[0]: Refreshing state... [id=i-019432b41727b66a0]
aws_instance.My-Instance-01[0]: Refreshing state... [id=i-0edc01737ec2fe49a]

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-01[0] will be destroyed
- resource "aws_instance" "My-Instance-01" {
  ami              = "ami-03f4878755434977f" -> null
  arn              = "arn:aws:ec2:ap-south-1:637423348062:instance/i-0edc01737ec2fe49a" -> null
  associate_public_ip_address = true -> null
  availability_zone = "ap-south-1b" -> null
  cpu_core_count   = 2 -> 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-0edc01737ec2fe49a" -> null
  instance_initiated_shutdown_behavior = "stop" -> null
  instance_state   = "running" -> null
  instance_type    = "t2.large" -> null
  ipv6_address_count = 0 -> null
  ipv6_addresses   = [] -> null
  monitoring       = false -> null
  placement_partition_number = 0 -> null
  primary_network_interface_id = "eni-04f93e4535aa36f0b" -> null
  private_dns      = "ip-172-31-15-150.ap-south-1.compute.internal" -> null
  private_ip       = "172.31.15.150" -> null
  public_dns       = "ec2-13-201-4-12.ap-south-1.compute.amazonaws.com" -> null
  public_ip        = "13.201.4.12" -> null
  secondary_private_ips = [] -> null
  security_groups  = [
    - "default",
  ] -> null
  source_dest_check = true -> null
  subnet_id        = "subnet-0e5f5e3d310ebacda" -> null
  tags             = {
    - "Name" = "UPES-EC2-Instnace"
  } -> null
  tags_all         = {
    - "Name" = "UPES-EC2-Instnace"
  } -> null
  tenancy          = "default" -> null
  user_data_replace_on_change = false -> null
  vpc_security_group_ids = [
    - "sg-0c6b5aae418c53ba2",
  ] -> null

  capacity_reservation_specification {

```

```
    - capacity_reservation_specification {
      - capacity_reservation_preference = "open" -> null
    }

  cpu_options {
    - core_count = 2 -> null
    - threads_per_core = 1 -> null
  }

  credit_specification {
    - cpu_credits = "standard" -> null
  }

  enclave_options {
    - enabled = false -> null
  }

  maintenance_options {
    - auto_recovery = "default" -> null
  }

  metadata_options {
    - http_endpoint = "enabled" -> null
    - http_protocol_ipv6 = "disabled" -> null
    - http_put_response_hop_limit = 1 -> null
    - http_tokens = "optional" -> null
    - instance_metadata_tags = "disabled" -> null
  }

  private_dns_name_options {
    - enable_resource_name_dns_a_record = false -> null
    - enable_resource_name_dns_aaaa_record = false -> null
    - hostname_type = "ip-name" -> null
  }

  root_block_device {
    - delete_on_termination = true -> null
    - device_name = "/dev/sdal" -> null
    - encrypted = false -> null
    - iops = 100 -> null
    - tags = {} -> null
    - throughput = 0 -> null
    - volume_id = "vol-834e2e0dc0826b5c9" -> null
    - volume_size = 8 -> null
    - volume_type = "gp2" -> null
  }
}

# aws_instance.My-Instance-02[0] will be destroyed
- resource "aws_instance" "My-Instance-02" {
  ami              = "ami-03f4878755434977f" -> null
  arn              = "arn:aws:ec2:ap-south-1:637423348062:instance/i-0513ee647c371165f" -> null
  associate_public_ip_address = true -> null
  availability_zone = "ap-south-1b" -> null
  cpu_core_count   = 2 -> null

```

```

- availability_zone = "ap-south-1b" -> null
- cpu_core_count = 2 -> 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-0513ee647c371165f" -> null
- instance_initiated_shutdown_behavior = "stop" -> null
- instance_state = "running" -> null
- instance_type = "t2.large" -> null
- ipv6_address_count = 0 -> null
- ipv6_addresses = [] -> null
- monitoring = false -> null
- placement_partition_number = 0 -> null
- primary_network_interface_id = "eni-0561ad241b40cc666" -> null
- private_dns = "ip-172-31-12-140.ap-south-1.compute.internal" -> null
- private_ip = "172.31.12.140" -> null
- public_dns = "ec2-13-235-49-48.ap-south-1.compute.amazonaws.com" -> null
- public_ip = "13.235.49.48" -> null
- secondary_private_ips = [] -> null
- security_groups = [
  - "default",
] -> null
- source_dest_check = true -> null
- subnet_id = "subnet-0e5f5e3d310ebacda" -> null
- tags = {
  - "Name" = "UPES-EC2-Instnace"
} -> null
- tags_all = {
  - "Name" = "UPES-EC2-Instnace"
} -> null
- tenancy = "default" -> null
- user_data_replace_on_change = false -> null
- vpc_security_group_ids = [
  - "sg-0c6b5aae418c53ba2",
] -> null

- capacity_reservation_specification {
  - capacity_reservation_preference = "open" -> null
}

- cpu_options {
  - core_count = 2 -> null
  - threads_per_core = 1 -> null
}

- credit_specification {
  - cpu_credits = "standard" -> null
}

- enclave_options {
  - enabled = false -> null
}

```

```

private_dns_name_options {
  - enable_resource_name_dns_a_record = false -> null
  - enable_resource_name_dns_aaaa_record = false -> null
  - hostname_type = "ip-name" -> null
}

root_block_device {
  - delete_on_termination = true -> null
  - device_name = "/dev/sda1" -> null
  - encrypted = false -> null
  - iops = 100 -> null
  - tags = {} -> null
  - throughput = 0 -> null
  - volume_id = "vol-0c01d690dac16551f" -> null
  - volume_size = 8 -> null
  - volume_type = "gp2" -> null
}

}

# aws_instance.My-Instnace-03[0] will be destroyed
- resource "aws_instance" "My-Instnace-03" {
  - ami = "ami-03f4878755434977f" -> null
  - arn = "arn:aws:ec2:ap-south-1:637423348062:instance/i-019432b41727b66a0" -> null
  - associate_public_ip_address = true -> null
  - availability_zone = "ap-south-1b" -> null
  - cpu_core_count = 2 -> 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-019432b41727b66a0" -> null
  - instance_initiated_shutdown_behavior = "stop" -> null
  - instance_state = "running" -> null
  - instance_type = "t2.large" -> null
  - ipv6_address_count = 0 -> null
  - ipv6_addresses = [] -> null
  - monitoring = false -> null
  - placement_partition_number = 0 -> null
  - primary_network_interface_id = "eni-052e91421ce0664f2" -> null
  - private_dns = "ip-172-31-5-134.ap-south-1.compute.internal" -> null
  - private_ip = "172.31.5.134" -> null
  - public_dns = "ec2-15-207-114-216.ap-south-1.compute.amazonaws.com" -> null
  - public_ip = "15.207.114.216" -> null
  - secondary_private_ips = [] -> null
  - security_groups = [
    - "default",
  ] -> null
  - source_dest_check = true -> null
  - subnet_id = "subnet-0e5f5e3d310ebacda" -> null
  - tags = {
    - "Name" = "UPES-EC2-Instnace"
  } -> null
  - tags_all = {

```

```

- cpu_options {
  - core_count      = 2 -> null
  - threads_per_core = 1 -> null
}

- credit_specification {
  - cpu_credits = "standard" -> null
}

- enclave_options {
  - enabled = false -> null
}

- maintenance_options {
  - auto_recovery = "default" -> null
}

- metadata_options {
  - http_endpoint           = "enabled" -> null
  - http_protocol_ipv6      = "disabled" -> null
  - http_put_response_hop_limit = 1 -> null
  - http_tokens             = "optional" -> null
  - instance_metadata_tags   = "disabled" -> null
}

- private_dns_name_options {
  - enable_resource_name_dns_a_record    = false -> null
  - enable_resource_name_dns_aaaa_record = false -> null
  - hostname_type                        = "ip-name" -> null
}

- root_block_device {
  - delete_on_termination = true -> null
  - device_name           = "/dev/sda1" -> null
  - encrypted             = false -> null
  - iops                  = 100 -> null
  - tags                  = {} -> null
  - throughput            = 0 -> null
  - volume_id             = "vol-093a5f5a63c87d1d5" -> null
  - volume_size           = 8 -> null
  - volume_type           = "gp2" -> null
}
}

```

Plan: 0 to add, 0 to change, 3 to destroy.

Do you really want to destroy all resources?

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-Instnace-02[0]: Destroying... [id=i-0513ee647c371165f]
aws_instance.My-Instnace-03[0]: Destroying... [id=i-019432b41727b66a0]
aws_instance.My-Instnace-01[0]: Destroying... [id=i-0edc01737ec2fe49a]

```

EC2 Dashboard

EC2 Global View

Events

Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Instances (6) info

Find Instance by attribute or tag (case-sensitive)

Any state

Connect

Instance state

Actions

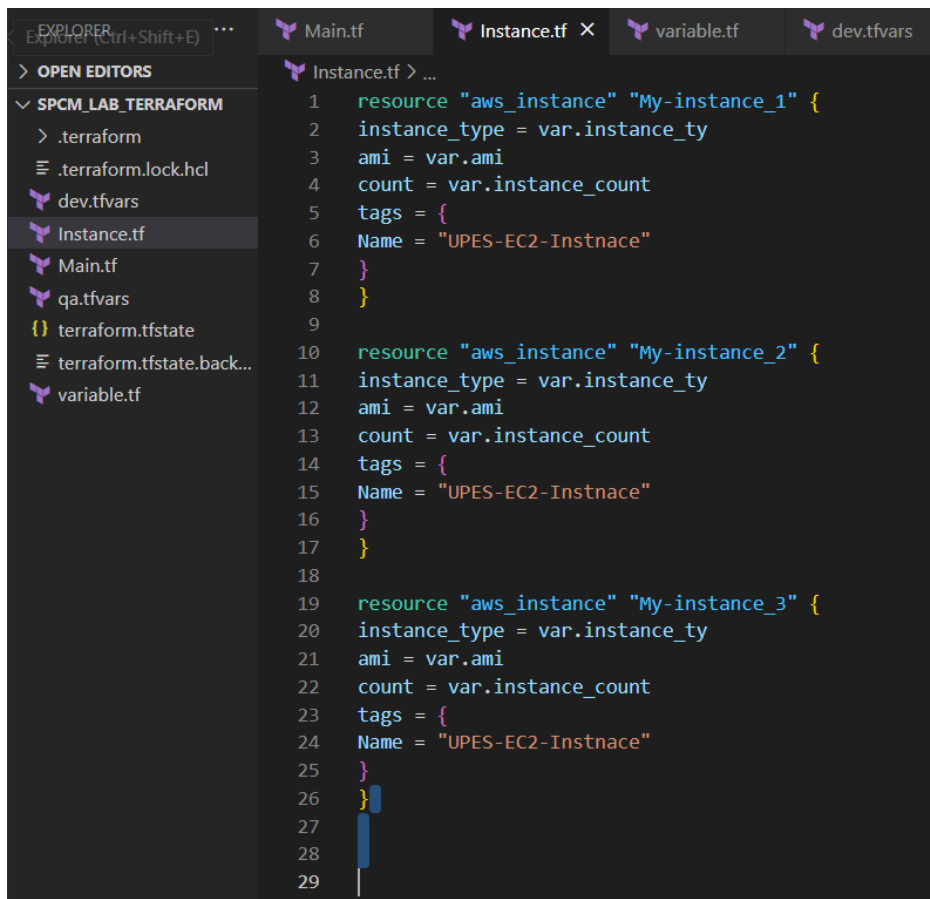
Launch instances

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	UPES-EC2-Inst...	i-0a2c04548c6185370	Terminated	t2.micro	-	View alarms	ap-south-1a
<input type="checkbox"/>	UPES-EC2-Inst...	i-0159152f0199b756a	Terminated	t2.micro	-	View alarms	ap-south-1a
<input type="checkbox"/>	UPES-EC2-Inst...	i-01c6be481182ff8cd	Terminated	t2.micro	-	View alarms	ap-south-1a
<input type="checkbox"/>	UPES-EC2-Inst...	i-0513ee647c371165f	Terminated	t2.large	-	View alarms	ap-south-1b
<input type="checkbox"/>	UPES-EC2-Inst...	i-019432b41727b66a0	Terminated	t2.large	-	View alarms	ap-south-1b
<input type="checkbox"/>	UPES-EC2-Inst...	i-0edc01737ec2fe49a	Terminated	t2.large	-	View alarms	ap-south-1b

LAB EXERCISE 6

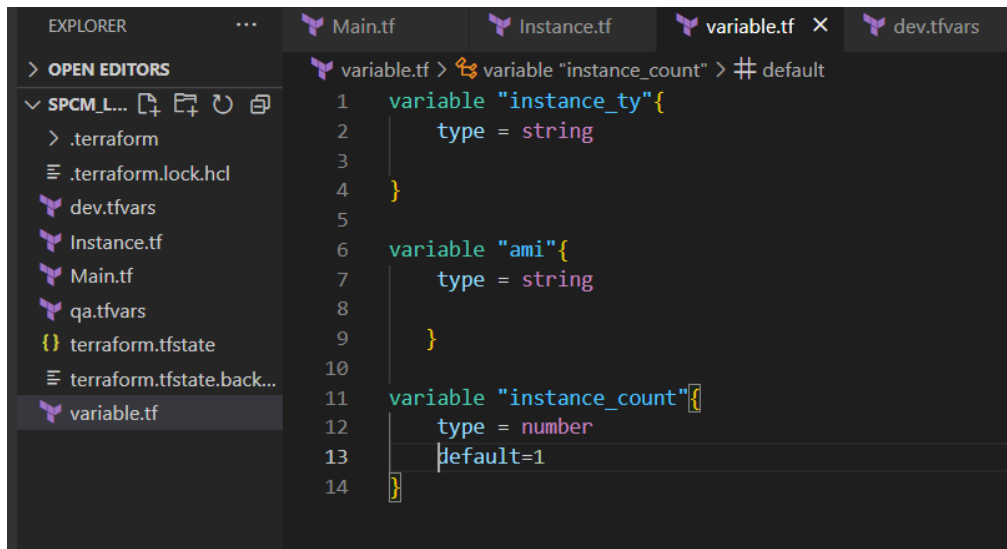
Aim: Terraform Multiple tfvars Files Objective:

Step 1: Create a instance.tf file

A screenshot of the Visual Studio Code editor interface. The Explorer sidebar on the left shows a project named 'SPCM_LAB_TERRAFORM' with files including '.terraform', '.terraform.lock.hcl', 'dev.tfvars', 'Instance.tf' (selected), 'Main.tf', 'qa.tfvars', 'terraform.tfstate', 'terraform.tfstate.back...', and 'variable.tf'. The main editor area displays the content of 'Instance.tf', which contains three Terraform resource blocks for 'aws_instance'. Each block defines 'My-instance_1', 'My-instance_2', and 'My-instance_3' with attributes for 'instance_type', 'ami', 'count', and 'tags'. The 'tags' attribute is a map with 'Name' set to 'UPES-EC2-Instnace'.

```
1 resource "aws_instance" "My-instance_1" {
2   instance_type = var.instance_ty
3   ami = var.ami
4   count = var.instance_count
5   tags = {
6     Name = "UPES-EC2-Instnace"
7   }
8 }
9
10 resource "aws_instance" "My-instance_2" {
11   instance_type = var.instance_ty
12   ami = var.ami
13   count = var.instance_count
14   tags = {
15     Name = "UPES-EC2-Instnace"
16   }
17 }
18
19 resource "aws_instance" "My-instance_3" {
20   instance_type = var.instance_ty
21   ami = var.ami
22   count = var.instance_count
23   tags = {
24     Name = "UPES-EC2-Instnace"
25   }
26 }
27
28
29
```

Step 2: Create a variable.tf file

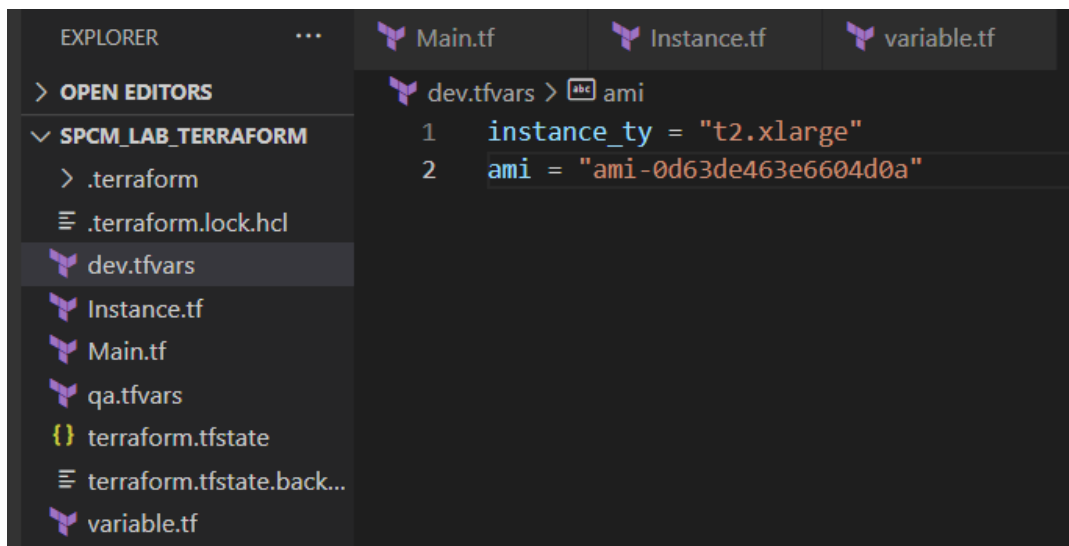


The screenshot shows the VS Code interface with the Explorer sidebar on the left and the variable.tf file open in the editor. The Explorer sidebar shows a project structure with folders like .terraform, .terraform.lock.hcl, and files like dev.tfvars, Instance.tf, Main.tf, qa.tfvars, terraform.tfstate, and terraform.tfstate.back... The variable.tf file contains the following Terraform code:

```
variable "instance_count" {  
  type = number  
  default = 1  
}  
  
variable "instance_ty" {  
  type = string  
}  
  
variable "ami" {  
  type = string  
}
```

Step 3: Create Multiple tfvars Files:

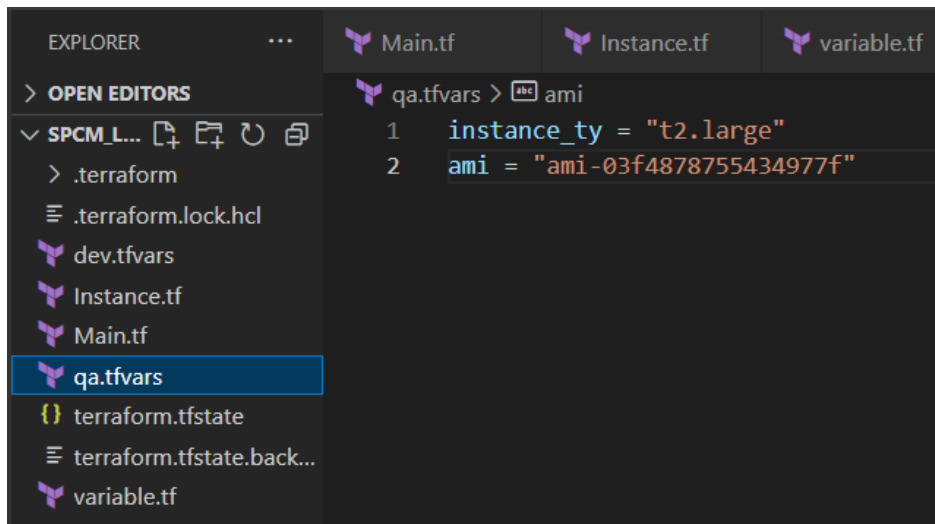
dev.tfvars



The screenshot shows the VS Code interface with the Explorer sidebar on the left and the dev.tfvars file open in the editor. The Explorer sidebar shows a project structure with folders like .terraform, .terraform.lock.hcl, and files like dev.tfvars, Instance.tf, Main.tf, qa.tfvars, terraform.tfstate, and terraform.tfstate.back... The dev.tfvars file contains the following Terraform variable assignments:

```
instance_ty = "t2.xlarge"  
ami = "ami-0d63de463e6604d0a"
```

qa.tfvars



Step 4: Now initializes

```
Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.31.0

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.
```

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform validate
Success! The configuration is valid.
```

Step 5: Apply for Dev Environment

```

F:\SEM 6\SPCH_LAB\SPCH_LAB_TERRAFORM>terraform apply -var-file=dev.tfvars

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-0d63de463e6604d0a"
  + 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.xlarge"
  + 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" = "UPES-EC2-Instnace"
  }
  + tags_all                = {

```

```

    + "Name" = "UPES-EC2-Instnace"
  }
  + 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)
}

# aws_instance.My-instance_2[0] will be created
+ resource "aws_instance" "My-instance_2" {
  + ami                    = "ami-0d63de463e6604d0a"
  + 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.xlarge"
  + 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" = "UPES-EC2-Instnace"
  }
  + tags_all                = {
    + "Name" = "UPES-EC2-Instnace"
  }
  + 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)
}

# aws_instance.My-instance_3[0] will be created
+ resource "aws_instance" "My-instance_3" {
  + ami                    = "ami-0d63de463e6604d0a"

```



```

aws_instance.My-instance_1[0]: Creating...
aws_instance.My-instance_3[0]: Creating...
aws_instance.My-instance_2[0]: Creating...
aws_instance.My-instance_3[0]: Still creating... [10s elapsed]
aws_instance.My-instance_1[0]: Still creating... [10s elapsed]
aws_instance.My-instance_2[0]: Still creating... [10s elapsed]
aws_instance.My-instance_3[0]: Creation complete after 14s [id=i-0c7c8f277790ae190]
aws_instance.My-instance_1[0]: Creation complete after 17s [id=i-07666f246d189f668]
aws_instance.My-instance_2[0]: Still creating... [20s elapsed]
aws_instance.My-instance_2[0]: Creation complete after 23s [id=i-0224bf2482e03e687]

```

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Platform
UPES-EC2-Inst...	i-0c7c8f277790ae190	Running	t2.xlarge	ⓘ Initializing	View alarms +	ap-south-1a	ec2
UPES-EC2-Inst...	i-0224bf2482e03e687	Running	t2.xlarge	ⓘ Initializing	View alarms +	ap-south-1a	ec2
UPES-EC2-Inst...	i-07666f246d189f668	Running	t2.xlarge	ⓘ Initializing	View alarms +	ap-south-1a	ec2

Step 6: Destroy Dev Environment

```

F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform destroy -var-file=dev.tfvars
aws_instance.My-instance_3[0]: Refreshing state... [id=i-0c7c8f277790ae190]
aws_instance.My-instance_1[0]: Refreshing state... [id=i-07666f246d189f668]
aws_instance.My-instance_2[0]: Refreshing state... [id=i-0224bf2482e03e687]

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-0d63de463e6684d0a" -> null
  - arn                   = "arn:aws:ec2:ap-south-1:637423348062:instance/i-07666f246d189f668" -> null
  - associate_public_ip_address = true -> null
  - availability_zone      = "ap-south-1a" -> null
  - cpu_core_count         = 4 -> 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-07666f246d189f668" -> null
  - instance_initiated_shutdown_behavior = "stop" -> null
  - instance_state         = "running" -> null
  - instance_type          = "t2.xlarge" -> null
  - ipv6_address_count      = 0 -> null
  - ipv6_addresses         = [] -> null
  - monitoring             = false -> null
  - placement_partition_number = 0 -> null
  - primary_network_interface_id = "eni-092f1b4b8b39306b9" -> null
  - private_dns            = "ip-172-31-35-201.ap-south-1.compute.internal" -> null
  - private_ip             = "172.31.35.201" -> null
  - public_dns             = "ec2-13-201-126-199.ap-south-1.compute.amazonaws.com" -> null
  - public_ip              = "13.201.126.199" -> null
  - secondary_private_ips   = [] -> null
  - security_groups        = [
    - "default",
  ] -> null
  - source_dest_check       = true -> null
  - subnet_id              = "subnet-0fb95688eaa188f7d" -> null
  - tags                    = {
    - "Name" = "UPES-EC2-Instnace"
  } -> null
  - tags_all               = {
    - "Name" = "UPES-EC2-Instnace"
  } -> null
  - tenancy                 = "default" -> null
}

```

```

- user_data_replace_on_change = false -> null
- vpc_security_group_ids = [
  - "sg-0c6b5aae418c53ba2",
] -> null

- capacity_reservation_specification {
  - capacity_reservation_preference = "open" -> null
}

- cpu_options {
  - core_count = 4 -> null
  - threads_per_core = 1 -> null
}

- credit_specification {
  - cpu_credits = "standard" -> null
}

- enclave_options {
  - enabled = false -> null
}

- maintenance_options {
  - auto_recovery = "default" -> null
}

- metadata_options {
  - http_endpoint = "enabled" -> null
  - http_protocol_ipv6 = "disabled" -> null
  - http_put_response_hop_limit = 2 -> null
  - http_tokens = "required" -> null
  - instance_metadata_tags = "disabled" -> null
}

- private_dns_name_options {
  - enable_resource_name_dns_a_record = false -> null
  - enable_resource_name_dns_aaaa_record = false -> null
  - hostname_type = "ip-name" -> null
}

- root_block_device {
  - delete_on_termination = true -> null
  - device_name = "/dev/xvda" -> null
  - encrypted = false -> null
  - iops = 3000 -> null
  - tags = {} -> null
  - throughput = 125 -> null
  - volume_id = "vol-0eb890ee6d0eb8c4a" -> null
  - volume_size = 8 -> null
  - volume_type = "gp3" -> null
}

```

```

}
}

# aws_instance.My-instance_2[0] will be destroyed
- resource "aws_instance" "My-instance_2" {
  - ami = "ami-0d63de463e6604d0a" -> null
  - arn = "arn:aws:ec2:ap-south-1:637423348062:instance/i-0224bf2482e03e687" -> null
  - associate_public_ip_address = true -> null
  - availability_zone = "ap-south-1a" -> null
  - cpu_core_count = 4 -> 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-0224bf2482e03e687" -> null
  - instance_initiated_shutdown_behavior = "stop" -> null
  - instance_state = "running" -> null
  - instance_type = "t2.xlarge" -> null
  - ipv6_address_count = 0 -> null
  - ipv6_addresses = [] -> null
  - monitoring = false -> null
  - placement_partition_number = 0 -> null
  - primary_network_interface_id = "eni-0092d5b00b8dcfb49" -> null
  - private_dns = "ip-172-31-34-41.ap-south-1.compute.internal" -> null
  - private_ip = "172.31.34.41" -> null
  - public_dns = "ec2-13-232-262-248.ap-south-1.compute.amazonaws.com" -> null
  - public_ip = "13.232.262.248" -> null
  - secondary_private_ips = [] -> null
  - security_groups = [
    - "default",
  ] -> null
  - source_dest_check = true -> null
  - subnet_id = "subnet-6fb9568eaa188f7d" -> null
  - tags = {
    - "Name" = "UPES-EC2-Instnace"
  } -> null
  - tags_all = {
    - "Name" = "UPES-EC2-Instnace"
  } -> null
  - tenancy = "default" -> null
  - user_data_replace_on_change = false -> null
  - vpc_security_group_ids = [
    - "sg-0c6b5aae418c53ba2",
  ] -> null
  - capacity_reservation_specification {
    - capacity_reservation_preference = "open" -> null
  }
}

```

```

- core_count          = 4 -> null
- threads_per_core    = 1 -> null
}

- credit_specification {
-   cpu_credits = "standard" -> null
}

- enclave_options {
-   enabled = false -> null
}

- maintenance_options {
-   auto_recovery = "default" -> null
}

- metadata_options {
-   http_endpoint          = "enabled" -> null
-   http_protocol_ipv6     = "disabled" -> null
-   http_put_response_hop_limit = 2 -> null
-   http_tokens            = "required" -> null
-   instance_metadata_tags = "disabled" -> null
}

- private_dns_name_options {
-   enable_resource_name_dns_a_record    = false -> null
-   enable_resource_name_dns_aaaa_record = false -> null
-   hostname_type                        = "ip-name" -> null
}

- root_block_device {
-   delete_on_termination = true -> null
-   device_name            = "/dev/xvda" -> null
-   encrypted              = false -> null
-   iops                   = 3000 -> null
-   tags                   = {} -> null
-   throughput             = 125 -> null
-   volume_id              = "vol-094b704f3be5d5220" -> null
-   volume_size            = 8 -> null
-   volume_type            = "gp3" -> null
}
}

```

Plan: 0 to add, 0 to change, 3 to destroy.

Do you really want to destroy all resources?

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_3[0]: Destroying... [id=i-0c7c8f277790ae190]
aws_instance.My-instance_2[0]: Destroying... [id=i-0224bf2482e03e687]
aws_instance.My-instance_1[0]: Destroying... [id=i-07666f246d189f668]
aws_instance.My-instance_1[0]: Still destroying... [id=i-07666f246d189f668, 10s elapsed]
aws_instance.My-instance_2[0]: Still destroying... [id=i-0224bf2482e03e687, 10s elapsed]
aws_instance.My-instance_3[0]: Still destroying... [id=i-0c7c8f277790ae190, 10s elapsed]
aws_instance.My-instance_2[0]: Still destroying... [id=i-0224bf2482e03e687, 21s elapsed]
aws_instance.My-instance_3[0]: Still destroying... [id=i-0c7c8f277790ae190, 21s elapsed]
aws_instance.My-instance_1[0]: Still destroying... [id=i-07666f246d189f668, 21s elapsed]
aws_instance.My-instance_3[0]: Still destroying... [id=i-0c7c8f277790ae190, 31s elapsed]
aws_instance.My-instance_2[0]: Still destroying... [id=i-0224bf2482e03e687, 31s elapsed]
aws_instance.My-instance_1[0]: Still destroying... [id=i-07666f246d189f668, 31s elapsed]
aws_instance.My-instance_2[0]: Destruction complete after 32s
aws_instance.My-instance_3[0]: Destruction complete after 32s
aws_instance.My-instance_1[0]: Destruction complete after 32s

```

Destroy complete! Resources: 3 destroyed.

EC2 Dashboard	Instances (3) Info		Connect	Instance state	Actions	Launch Instances
EC2 Global View	Find Instance by attribute or tag (case-sensitive)		Any state	< 1 >		
Events	<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check
▼ Instances	<input type="checkbox"/>	UPES-EC2-Inst...	i-0c7c8f277790ae190	Terminated	t2.xlarge	View alarms +
Instances	<input type="checkbox"/>	UPES-EC2-Inst...	i-0224bf2482e03e687	Terminated	t2.xlarge	View alarms +
Instance Types	<input type="checkbox"/>	UPES-EC2-Inst...	i-07666f246d189f668	Terminated	t2.xlarge	View alarms +
Launch Templates						

Step 7: Apply for Qa Environment

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform apply -var-file=qa.tfvars

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-03f4878755434977f"
  + 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.large"
  + 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" = "UPES-EC2-Instnace"
  }
  + tags_all                    = {

```

```

    + "Name" = "UPES-EC2-Instnace"
  }
  + 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)
}

# aws_instance.My-instance_2[0] will be created
+ resource "aws_instance" "My-instance_2" {
  + ami                         = "ami-03f4878755434977f"
  + 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.large"
  + 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" = "UPES-EC2-Instnace"
  }
  + tags_all                    = {

```

```

+ 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)
}

# aws_instance.My-instance_3[0] will be created
+ resource "aws_instance" "My-instance_3" {
  + ami = "ami-03f4878755434977f"
  + 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.large"
  + 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" = "UPES-EC2-Instnace"
  }
  + tags_all = {
    + "Name" = "UPES-EC2-Instnace"
  }
  + 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)
}

```

EC2 Dashboard

EC2 Global View

Events

Instances

Instances Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Instances (6) Info

Find instance by attribute or tag (case-sensitive)

Any state

Connect

Instance state

Actions

Launch Instances

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Put
UPES-EC2-Inst...	i-0c7d8f277790ae190	Terminated	t2.large	-	View alarms +	ap-south-1a	-
UPES-EC2-Inst...	i-02240f2402403e687	Terminated	t2.xlarge	-	View alarms +	ap-south-1a	-
UPES-EC2-Inst...	i-07660246d189f668	Terminated	t2.xlarge	-	View alarms +	ap-south-1a	-
UPES-EC2-Inst...	i-06c77b7854044392	Running	t2.large	⊙ Initializing	View alarms +	ap-south-1b	ec2
UPES-EC2-Inst...	i-0b487cae8f8e5266	Running	t2.large	⊙ Initializing	View alarms +	ap-south-1b	ec2
UPES-EC2-Inst...	i-02856e705a796951b	Running	t2.large	⊙ Initializing	View alarms +	ap-south-1b	ec2

Step 8: Destroy for Qa Environment

```

F:\SER\G\SPCH\LAB\SPCH\LAB_TERRAFORM>terraform destroy -var=files=tfvars
aws_instance.My-instance_3[0]: Refreshing state... [id=i-02656e70b0d96951b]
aws_instance.My-instance_1[0]: Refreshing state... [id=i-06c77fb7850404392]
aws_instance.My-instance_2[0]: Refreshing state... [id=i-0b07cae8f06e2260]

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-0346878750434977f" -> null
  - arm                        = "arn:aws:ec2:ap-south-1:637423348862:instance/i-06c77fb7850404392" -> null
  - associate_public_ip_address = true -> null
  - availability_zone           = "ap-south-1b" -> null
  - cpu_core_count              = 2 -> 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-06c77fb7850404392" -> null
  - instance_initiated_shutdown_behavior = "stop" -> null
  - instance_state              = "running" -> null
  - instance_type               = "t2.large" -> null
  - ipv4_address_count          = 0 -> null
  - ipv4_addresses              = [] -> null
  - monitoring                  = false -> null
  - placement_partition_number = 0 -> null
  - primary_network_interface_id = "eni-0186d9b707a770605" -> null
  - private_dns                 = "ip-172-31-0-0.ap-south-1.compute.internal" -> null
  - private_ip                  = "172.31.0.9" -> null
  - public_dns                  = "ec2-172-31-0-109.ap-south-1.compute.amazonaws.com" -> null
  - public_ip                   = "54.199.122.109" -> null
  - secondary_private_ips       = [] -> null
  - security_groups              = [
    - "default",
  ] -> null
  - source_dest_check           = true -> null
  - subnet_id                   = "subnet-de5f5d3d110ebacda" -> null
  - tags                         = {
    - "Name" = "IPES-EC2-Instance"
  } -> null
  - tags_all                    = {
    - "Name" = "IPES-EC2-Instance"
  } -> null
  - tenancy                     = "default" -> null
  - user_data_replace_on_change = false -> null
  - vpc_security_group_ids      = [
    - "sg-0c6b5aae418c53ba2",
  ] -> null
  - capacity_reservation_specification {
    - capacity_reservation_preference = "open" -> null
  }
  - cpu_options {
    - core_count          = 2 -> null
    - threads_per_core    = 1 -> null
  }
  - credit_specification {
    - cpu_credits = "standard" -> null
  }
  - enclave_options {
    - enabled = false -> null
  }
  - maintenance_options {
    - auto_recovery = "default" -> null
  }
  - metadata_options {
    - http_endpoint          = "enabled" -> null
    - http_protocol_ipv6    = "disabled" -> null
    - http_put_response_hop_limit = 1 -> null
    - http_tokens            = "optional" -> null
    - instance_metadata_tags = "disabled" -> null
  }
  - private_dns_name_options {
    - enable_resource_name_dns_a_record    = false -> null
    - enable_resource_name_dns_aaaa_record = false -> null
    - hostname_type                        = "ip-name" -> null
  }
  - root_block_device {
    - delete_on_termination = true -> null
    - device_name           = "/dev/sda1" -> null
    - encrypted             = false -> null
    - iops                  = 100 -> null
    - tags                  = {} -> null
    - throughput            = 0 -> null
    - volume_id             = "vol-06344da595e19b341c" -> null
    - volume_size           = 8 -> null
    - volume_type           = "gp2" -> null
  }
}

# aws_instance.My-instance_2[0] will be destroyed
- resource "aws_instance" "My-instance_2" {

```

```

- tenancy                     = "default" -> null
- user_data_replace_on_change = false -> null
- vpc_security_group_ids      = [
  - "sg-0c6b5aae418c53ba2",
] -> null

- capacity_reservation_specification {
  - capacity_reservation_preference = "open" -> null
}

- cpu_options {
  - core_count          = 2 -> null
  - threads_per_core    = 1 -> null
}

- credit_specification {
  - cpu_credits = "standard" -> null
}

- enclave_options {
  - enabled = false -> null
}

- maintenance_options {
  - auto_recovery = "default" -> null
}

- metadata_options {
  - http_endpoint          = "enabled" -> null
  - http_protocol_ipv6    = "disabled" -> null
  - http_put_response_hop_limit = 1 -> null
  - http_tokens            = "optional" -> null
  - instance_metadata_tags = "disabled" -> null
}

- private_dns_name_options {
  - enable_resource_name_dns_a_record    = false -> null
  - enable_resource_name_dns_aaaa_record = false -> null
  - hostname_type                        = "ip-name" -> null
}

- root_block_device {
  - delete_on_termination = true -> null
  - device_name           = "/dev/sda1" -> null
  - encrypted             = false -> null
  - iops                  = 100 -> null
  - tags                  = {} -> null
  - throughput            = 0 -> null
  - volume_id             = "vol-06344da595e19b341c" -> null
  - volume_size           = 8 -> null
  - volume_type           = "gp2" -> null
}

# aws_instance.My-instance_2[0] will be destroyed
- resource "aws_instance" "My-instance_2" {

```

```

- core_count      = 2 -> null
- threads_per_core = 1 -> null
}

- credit_specification {
-   cpu_credits = "standard" -> null
}

- enclave_options {
-   enabled = false -> null
}

- maintenance_options {
-   auto_recovery = "default" -> null
}

- metadata_options {
-   http_endpoint      = "enabled" -> null
-   http_protocol_ipv6 = "disabled" -> null
-   http_put_response_hop_limit = 1 -> null
-   http_tokens        = "optional" -> null
-   instance_metadata_tags = "disabled" -> null
}

- private_dns_name_options {
-   enable_resource_name_dns_a_record   = false -> null
-   enable_resource_name_dns_aaaa_record = false -> null
-   hostname_type                       = "ip-name" -> null
}

- root_block_device {
-   delete_on_termination = true -> null
-   device_name            = "/dev/sda1" -> null
-   encrypted              = false -> null
-   iops                   = 100 -> null
-   tags                   = {} -> null
-   throughput             = 0 -> null
-   volume_id              = "vol-0afbbb2fbd6ece80d" -> null
-   volume_size            = 8 -> null
-   volume_type            = "gp2" -> null
}
}

# aws_instance.My-instance_3[0] will be destroyed
- resource "aws_instance" "My-instance_3" {
-   ami                        = "ami-03f4878755434977f" -> null
-   arn                      = "arn:aws:ec2:ap-south-1:637423348062:instance/i-02656e705a096951b" -> null
-   associate_public_ip_address = true -> null
-   availability_zone         = "ap-south-1b" -> null
-   cpu_core_count            = 2 -> 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

```

```

}

- metadata_options {
  - http_endpoint          = "enabled" -> null
  - http_protocol_ipv6     = "disabled" -> null
  - http_put_response_hop_limit = 1 -> null
  - http_tokens            = "optional" -> null
  - instance_metadata_tags  = "disabled" -> null
}

- private_dns_name_options {
  - enable_resource_name_dns_a_record = false -> null
  - enable_resource_name_dns_aaaa_record = false -> null
  - hostname_type                    = "ip-name" -> null
}

- root_block_device {
  - delete_on_termination = true -> null
  - device_name           = "/dev/sdal" -> null
  - encrypted             = false -> null
  - iops                  = 100 -> null
  - tags                  = {} -> null
  - throughput            = 0 -> null
  - volume_id             = "vol-010656a1835c8dbff" -> null
  - volume_size           = 8 -> null
  - volume_type           = "gp2" -> null
}
}

```

Plan: 0 to add, 0 to change, 3 to destroy.

Do you really want to destroy all resources?

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_2[0]: Destroying... [id=i-0bd87cae8f08e5266]
aws_instance.My-instance_3[0]: Destroying... [id=i-02656e705a096951b]
aws_instance.My-instance_1[0]: Destroying... [id=i-06c77fb7854044392]
aws_instance.My-instance_1[0]: Still destroying... [id=i-06c77fb7854044392, 10s elapsed]
aws_instance.My-instance_3[0]: Still destroying... [id=i-02656e705a096951b, 10s elapsed]
aws_instance.My-instance_2[0]: Still destroying... [id=i-0bd87cae8f08e5266, 10s elapsed]
aws_instance.My-instance_2[0]: Still destroying... [id=i-0bd87cae8f08e5266, 20s elapsed]
aws_instance.My-instance_3[0]: Still destroying... [id=i-02656e705a096951b, 20s elapsed]
aws_instance.My-instance_1[0]: Still destroying... [id=i-06c77fb7854044392, 20s elapsed]
aws_instance.My-instance_2[0]: Still destroying... [id=i-0bd87cae8f08e5266, 30s elapsed]
aws_instance.My-instance_1[0]: Still destroying... [id=i-06c77fb7854044392, 30s elapsed]
aws_instance.My-instance_3[0]: Still destroying... [id=i-02656e705a096951b, 30s elapsed]
aws_instance.My-instance_2[0]: Destruction complete after 33s
aws_instance.My-instance_1[0]: Destruction complete after 33s
aws_instance.My-instance_3[0]: Destruction complete after 33s

```

Destroy complete! Resources: 3 destroyed.

EC2 Dashboard

EC2 Global View

Events

Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Instances (6) info

Find Instance by attribute or tag (case-sensitive)

Any state

Connect

Instance state

Actions

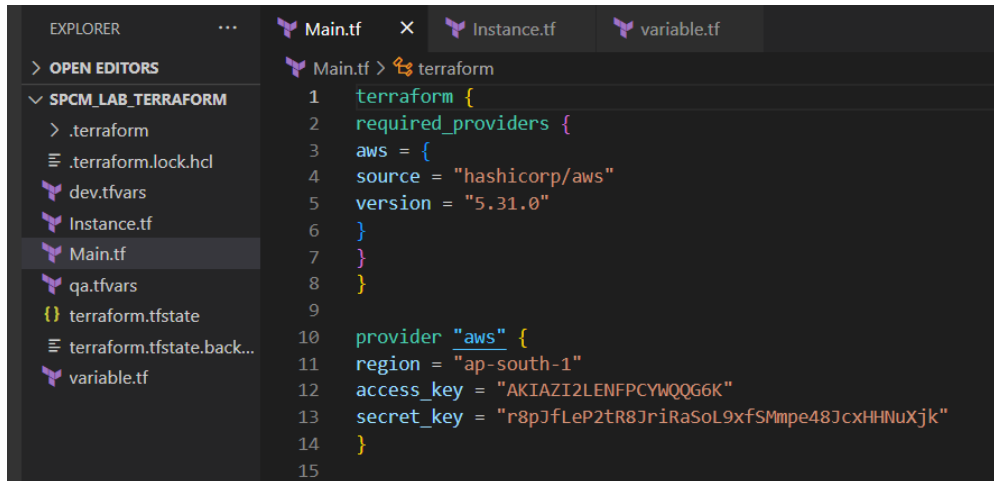
Launch instances

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	UPES-EC2-Inst...	i-0c7c8f277790ae190	Terminated	t2.xlarge	-	View alarms	ap-south-1a
<input type="checkbox"/>	UPES-EC2-Inst...	i-0224bf2482e03e687	Terminated	t2.xlarge	-	View alarms	ap-south-1a
<input type="checkbox"/>	UPES-EC2-Inst...	i-07666f246d189f668	Terminated	t2.xlarge	-	View alarms	ap-south-1a
<input type="checkbox"/>	UPES-EC2-Inst...	i-06c77fb7854044392	Terminated	t2.large	-	View alarms	ap-south-1b
<input type="checkbox"/>	UPES-EC2-Inst...	i-0bd87cae8f08e5266	Terminated	t2.large	-	View alarms	ap-south-1b
<input type="checkbox"/>	UPES-EC2-Inst...	i-02656e705a096951b	Terminated	t2.large	-	View alarms	ap-south-1b

LAB EXERCISE 7

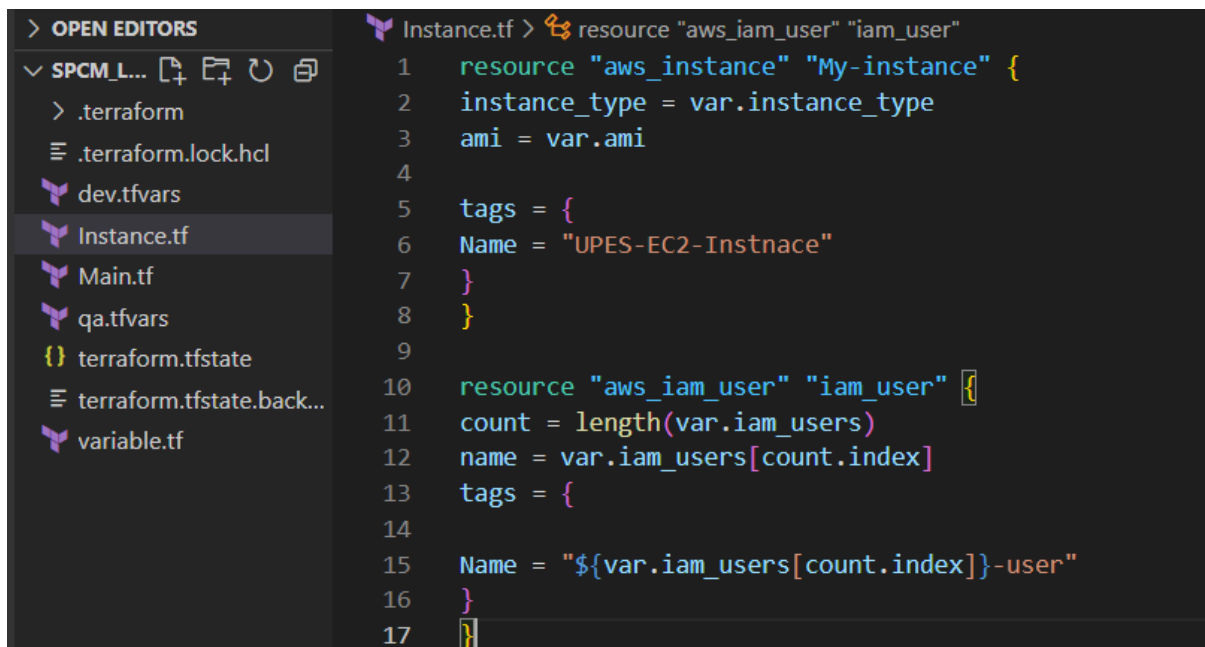
Aim: Creating Multiple IAM Users in Terraform

Step 1: Create a main.tf file.

A screenshot of the Visual Studio Code editor interface. The Explorer sidebar on the left shows a project named 'SPCM_LAB_TERRAFORM' with a '.terraform' directory containing files like '.terraform.lock.hcl', 'dev.tfvars', 'Instance.tf', 'Main.tf' (selected), 'qa.tfvars', 'terraform.tfstate', and 'terraform.tfstate.back...'. The main editor area shows the 'Main.tf' file with Terraform configuration for the AWS provider. The code is as follows:

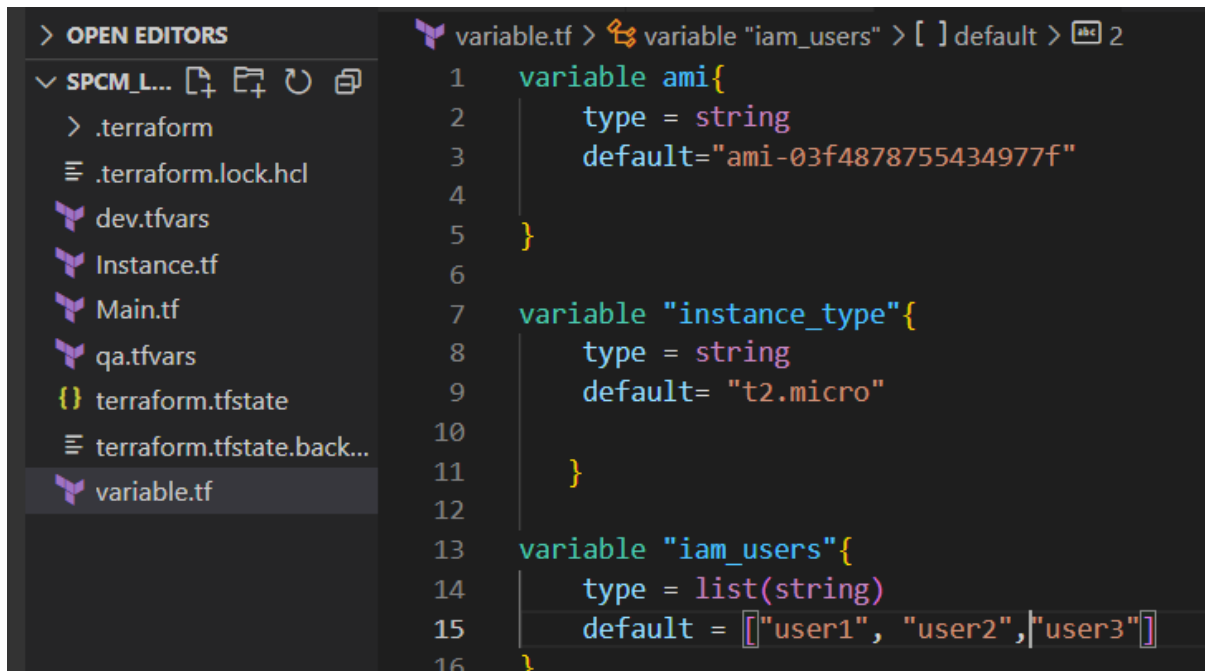
```
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "5.31.0"
6     }
7   }
8 }
9
10 provider "aws" {
11   region = "ap-south-1"
12   access_key = "AKIAZI2LENFPCYWQQG6K"
13   secret_key = "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"
14 }
15
```

Step 2: Create a instance.tf file

A screenshot of the Visual Studio Code editor interface. The Explorer sidebar on the left shows the same project structure as before, with 'Instance.tf' now selected. The main editor area shows the 'Instance.tf' file with Terraform configuration for creating an EC2 instance and an IAM user. The code is as follows:

```
1 resource "aws_instance" "My-instance" {
2   instance_type = var.instance_type
3   ami = var.ami
4
5   tags = {
6     Name = "UPES-EC2-Instnace"
7   }
8 }
9
10 resource "aws_iam_user" "iam_user" {
11   count = length(var.iam_users)
12   name = var.iam_users[count.index]
13   tags = {
14
15     Name = "${var.iam_users[count.index]}-user"
16   }
17 }
```

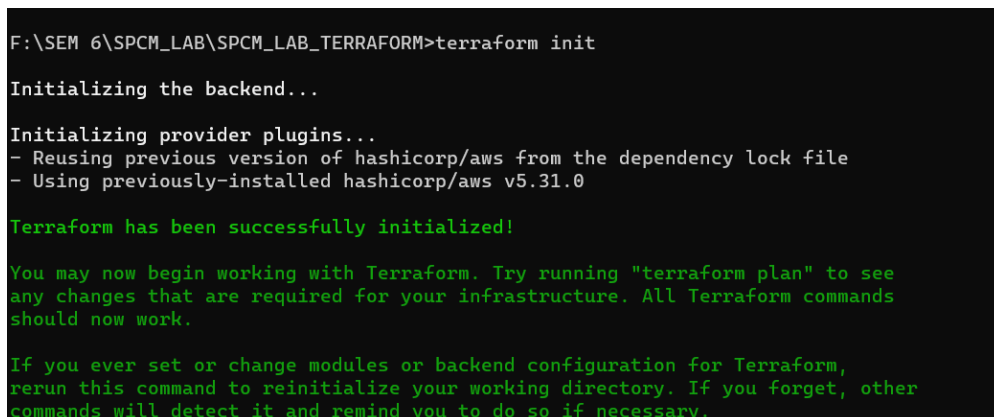
Step 3: Create a variable.tf file



The screenshot shows a code editor with a sidebar on the left and a main editor area on the right. The sidebar, titled 'OPEN EDITORS', lists several files: .terraform, .terraform.lock.hcl, dev.tfvars, Instance.tf, Main.tf, qa.tfvars, terraform.tfstate, terraform.tfstate.back..., and variable.tf. The 'variable.tf' file is selected and highlighted. The main editor area displays the content of 'variable.tf' with line numbers 1 through 16. The code defines three variables: 'ami' (a string with a default value), 'instance_type' (a string with a default value), and 'iam_users' (a list of strings with a default value).

```
1 variable ami{
2     type = string
3     default="ami-03f4878755434977f"
4 }
5
6
7 variable "instance_type"{
8     type = string
9     default= "t2.micro"
10 }
11
12
13 variable "iam_users"{
14     type = list(string)
15     default = ["user1", "user2", "user3"]
16 }
```

Step 4: Now initializes



The screenshot shows a terminal window with the following text:

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform init

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.31.0

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.
```

Step 5: Now perform validate



The screenshot shows a terminal window with the following text:

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform validate
Success! The configuration is valid.
```

Step 6: Now perform the terraform apply

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform validate
Success! The configuration is valid.

F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform apply

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_iam_user.iam_user[0] will be created
+ resource "aws_iam_user" "iam_user" {
+   arn          = (known after apply)
+   force_destroy = false
+   id           = (known after apply)
+   name         = "user1"
+   path         = "/"
+   tags         = {
+     "Name" = "user1-user"
+   }
+   tags_all     = {
+     "Name" = "user1-user"
+   }
+   unique_id    = (known after apply)
}

# aws_iam_user.iam_user[1] will be created
+ resource "aws_iam_user" "iam_user" {
+   arn          = (known after apply)
+   force_destroy = false
+   id           = (known after apply)
+   name         = "user2"
+   path         = "/"
+   tags         = {
+     "Name" = "user2-user"
+   }
+   tags_all     = {
+     "Name" = "user2-user"
+   }
+   unique_id    = (known after apply)
}

# aws_iam_user.iam_user[2] will be created
+ resource "aws_iam_user" "iam_user" {
+   arn          = (known after apply)
+   force_destroy = false
+   id           = (known after apply)
+   name         = "user3"
+   path         = "/"
+   tags         = {
+     "Name" = "user3-user"
+   }
+   tags_all     = {
+     "Name" = "user3-user"
+   }
+   unique_id    = (known after apply)
}

# aws_instance.My-instance will be created
+ resource "aws_instance" "My-instance" {
```

Instances (1) Info							
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>				<input type="text" value="Any state"/>		<input type="button" value="Launch instances"/>	
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	UPES-EC2-Inst...	i-0d7b168226bb58756	Running	t2.micro	Initializing	View alarms	ap-south-1a

Step 7: Now perform Terraform destroy

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform destroy

aws_iam_user.iam_user[1]: Refreshing state... [id=user2]
aws_iam_user.iam_user[2]: Refreshing state... [id=user3]
aws_iam_user.iam_user[0]: Refreshing state... [id=user1]
aws_instance.My-instance: Refreshing state... [id=i-0d7b168226bb58756]

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_iam_user.iam_user[0] will be destroyed
- resource "aws_iam_user" "iam_user" {
-   arn          = "arn:aws:iam::637423348862:user/user1" -> null
-   force_destroy = false -> null
-   id           = "user1" -> null
-   name         = "user1" -> null
-   path         = "/" -> null
-   tags         = {
-     "Name" = "user1-user"
-   } -> null
-   tags_all     = {
-     "Name" = "user1-user"
-   } -> null
-   unique_id    = "AIDAZ12LENFPD1YHG9EM" -> null
}

# aws_iam_user.iam_user[1] will be destroyed
- resource "aws_iam_user" "iam_user" {
-   arn          = "arn:aws:iam::637423348862:user/user2" -> null
-   force_destroy = false -> null
-   id           = "user2" -> null
-   name         = "user2" -> null
-   path         = "/" -> null
-   tags         = {
-     "Name" = "user2-user"
-   } -> null
-   tags_all     = {
-     "Name" = "user2-user"
-   } -> null
-   unique_id    = "AIDAZ12LENFPA0B70EGDA" -> null
}

# aws_iam_user.iam_user[2] will be destroyed
- resource "aws_iam_user" "iam_user" {
-   arn          = "arn:aws:iam::637423348862:user/user3" -> null
-   force_destroy = false -> null
-   id           = "user3" -> null
-   name         = "user3" -> null
-   path         = "/" -> null
-   tags         = {
-     "Name" = "user3-user"
-   } -> null
-   tags_all     = {
-     "Name" = "user3-user"
-   } -> null
-   unique_id    = "AIDAZ12LENFPE3G0W0NZ" -> null
}

# aws_instance.My-instance will be destroyed
- resource "aws_instance" "My-instance" {
-   ami          = "ami-03f4878755434977f" -> null
-   arn          = "arn:aws:ec2:ap-south-1:637423348862:instance/i-0d7b168226bb58756" -> null
}
```

```

- get_password_data           = false -> null
- hibernation                 = false -> null
- id                         = "i-0d7b168226bb58756" -> 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-0a138d0d0dc7f8bc" -> null
- private_dns                 = "ip-172-31-37-220.ap-south-1.compute.internal" -> null
- private_ip                  = "172.31.37.220" -> null
- public_dns                  = "ec2-43-205-230-157.ap-south-1.compute.amazonaws.com" -> null
- public_ip                   = "43.205.230.157" -> null
- secondary_private_ips       = [] -> null
- security_groups              = [
  - "default",
] -> null
- source_dest_check           = true -> null
- subnet_id                   = "subnet-0fb95688eaa188f7d" -> null
- tags                         = {
  "Name" = "UPES-EC2-Instnace"
} -> null
- tags_all                    = {
  "Name" = "UPES-EC2-Instnace"
} -> null
- tenancy                     = "default" -> null
- user_data_replace_on_change = false -> null
- vpc_security_group_ids      = [
  "sg-bcbbbaae18c3ba2",
] -> null
- capacity_reservation_specification {
  - capacity_reservation_preference = "open" -> null
}
- cpu_options {
  - core_count       = 1 -> null
  - threads_per_core = 1 -> null
}
- credit_specification {
  - cpu_credits = "standard" -> null
}
- enclave_options {
  - enabled = false -> null
}
- maintenance_options {
  - auto_recovery = "default" -> null
}
- metadata_options {
  - http_endpoint           = "enabled" -> null
  - http_protocol_ipv6      = "disabled" -> null
  - http_put_response_hop_limit = 1 -> null
  - http_tokens              = "optional" -> null
  - instance_metadata_tags   = "disabled" -> null
}
- private_dns_name_options {
  - enable_resource_name_dns_a_record = false -> null

```

```

}
- credit_specification {
  - cpu_credits = "standard" -> null
}
- enclave_options {
  - enabled = false -> null
}
- maintenance_options {
  - auto_recovery = "default" -> null
}
- metadata_options {
  - http_endpoint           = "enabled" -> null
  - http_protocol_ipv6      = "disabled" -> null
  - http_put_response_hop_limit = 1 -> null
  - http_tokens              = "optional" -> null
  - instance_metadata_tags   = "disabled" -> null
}
- private_dns_name_options {
  - enable_resource_name_dns_a_record = false -> null
  - enable_resource_name_dns_aaaa_record = false -> null
  - hostname_type                     = "ip-name" -> null
}
- root_block_device {
  - delete_on_termination = true -> null
  - device_name           = "/dev/sda1" -> null
  - encrypted              = false -> null
  - iops                   = 100 -> null
  - tags                   = {} -> null
  - throughput             = 0 -> null
  - volume_id              = "vol-08fdb35ff14989557" -> null
  - volume_size            = 8 -> null
  - volume_type            = "gp2" -> null
}
}

```

Plan: 0 to add, 0 to change, 4 to destroy.

Do you really want to destroy all resources?
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_iam_user.iam_user[0]: Destroying... [id=user1]
aws_iam_user.iam_user[1]: Destroying... [id=user2]
aws_iam_user.iam_user[2]: Destroying... [id=user3]
aws_instance.My-instance: Destroying... [id=i-0d7b168226bb58756]
aws_iam_user.iam_user[0]: Destruction complete after 2s
aws_iam_user.iam_user[1]: Destruction complete after 2s
aws_iam_user.iam_user[2]: Destruction complete after 2s
aws_instance.My-instance: Still destroying... [id=i-0d7b168226bb58756, 10s elapsed]
aws_instance.My-instance: Still destroying... [id=i-0d7b168226bb58756, 20s elapsed]
aws_instance.My-instance: Still destroying... [id=i-0d7b168226bb58756, 30s elapsed]
aws_instance.My-instance: Destruction complete after 31s

```

Destroy complete! Resources: 4 destroyed.

Instances (1) Info							
Find Instance by attribute or tag (case-sensitive)				Any state		< 1 >	
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	UPES-EC2-Inst...	i-0d7b168226bb58756	Terminated	t2.micro	-	View alarms	ap-south-1a