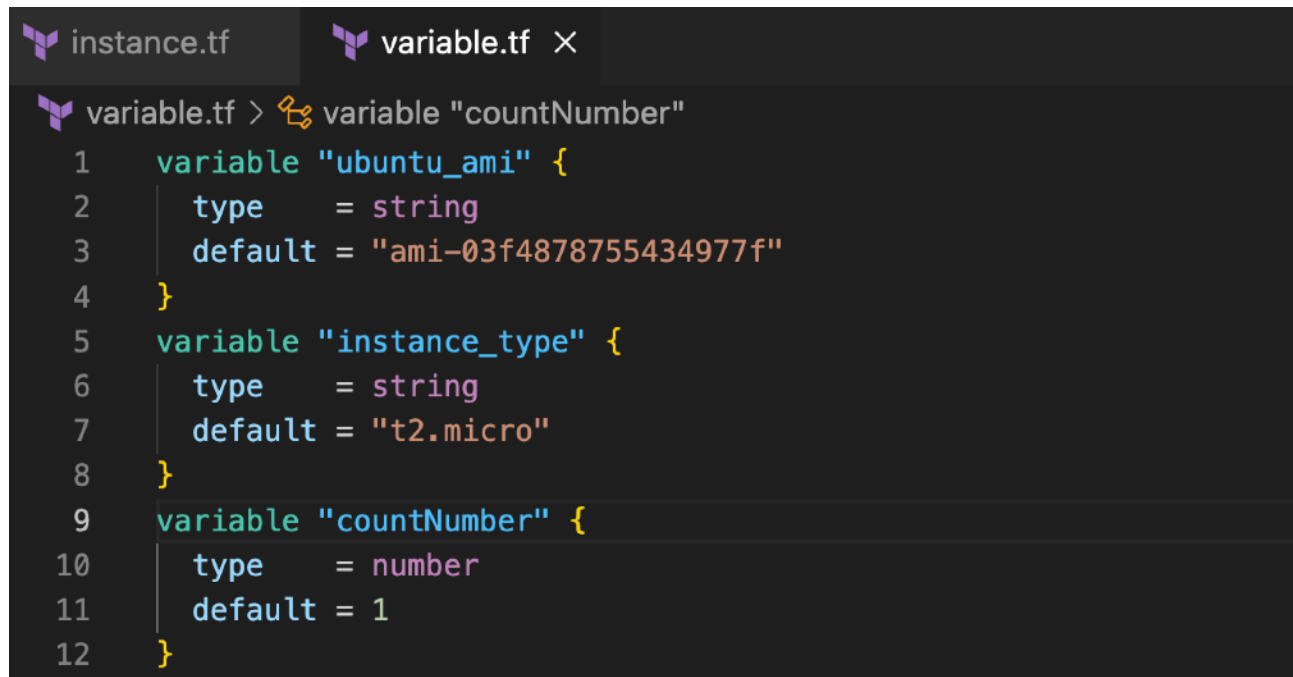# SPCM LAB-4

Objective: Learn how to define and use variables in Terraform configuration.
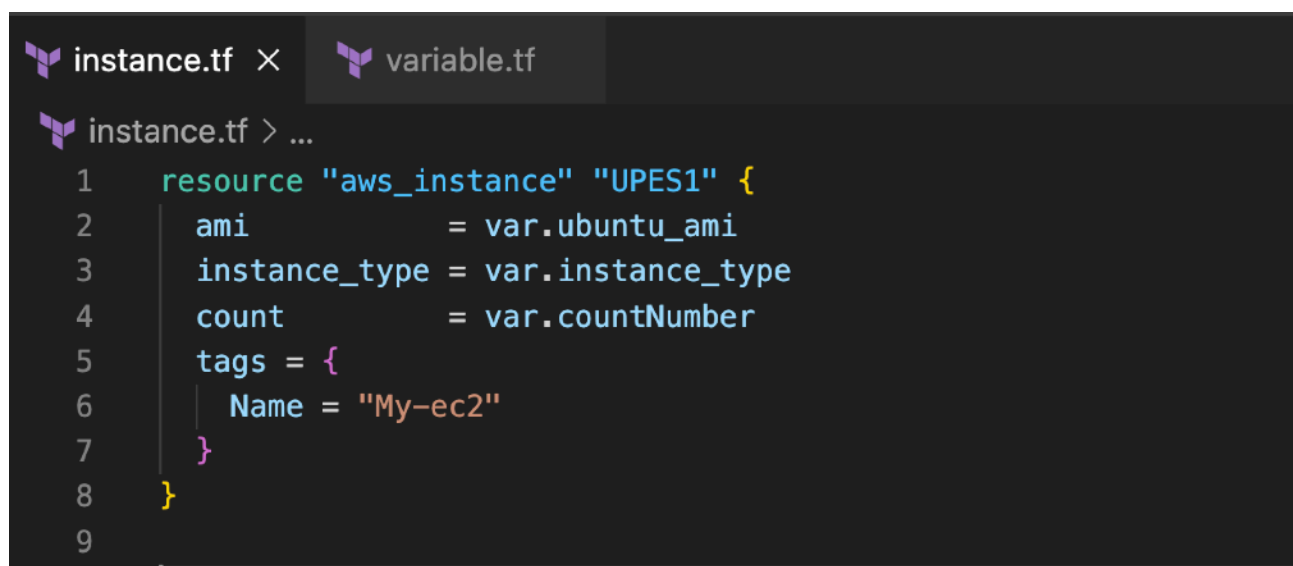
• Create a file variable.tf with the following contents.

```
gauravbhandari@gauravs-MacBook-Air-2 aws-terraform-demo % touch variable.tf
```

```
variable "ubuntu_ami" {
  type    = string
  default = "ami-03f4878755434977f"
}
variable "instance_type" {
  type    = string
  default = "t2.micro"
}
variable "countNumber" {
  type    = number
  default = 1
}
```

• Use the variable declared and defined in variable.tf in instance.tf

```
resource "aws_instance" "UPES1" {
  ami           = var.ubuntu_ami
  instance_type = var.instance_type
  count         = var.countNumber
  tags = {
    Name = "My-ec2"
  }
}
```

- Run the command terraform plan and review the plan to see if it meets your expectations

```
gauravbhandari@gauravs-MacBook-Air-2 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.UPES1[0] will be created
  + resource "aws_instance" "UPES1" {
      + ami                          = "ami-03f4878755434977f"
      + arn                          = (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.
```

- Run terraform apply and create the resources.

```
gauravbhandari@gauravs-MacBook-Air-2 aws-terraform-demo % terraform apply --auto-approve

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

Terraform will perform the following actions:

  # aws_instance.UPES1[0] will be created
  + resource "aws_instance" "UPES1" {
      + ami                          = "ami-03f4878755434977f"
```

```
Plan: 1 to add, 0 to change, 0 to destroy.
aws_instance.UPES1[0]: Creating...
aws_instance.UPES1[0]: Still creating... [10s elapsed]
aws_instance.UPES1[0]: Still creating... [20s elapsed]
aws_instance.UPES1[0]: Still creating... [30s elapsed]
aws_instance.UPES1[0]: Creation complete after 33s [id=i-07be4acb91cff5340]

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

- When you are done experimenting, run the following command to destroy the createdre sources.

```
gauravbhandari@gauravs-Air-2 aws-terraform-demo % terraform destroy --auto-approve
aws_instance.UPES1[0]: Refreshing state... [id=i-07be4acb91cff5340]

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

Terraform will perform the following actions:
```

```
Plan: 0 to add, 0 to change, 1 to destroy.
aws_instance.UPES1[0]: Destroying... [id=i-07be4acb91cff5340]
aws_instance.UPES1[0]: Still destroying... [id=i-07be4acb91cff5340, 10s elapsed]
aws_instance.UPES1[0]: Still destroying... [id=i-07be4acb91cff5340, 20s elapsed]
aws_instance.UPES1[0]: Still destroying... [id=i-07be4acb91cff5340, 30s elapsed]
aws_instance.UPES1[0]: Destruction complete after 31s

Destroy complete! Resources: 1 destroyed.
```

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pub
☐	My-ec2	i-07be4acb91cff5340	⊖ Terminated	t2.micro	–	View alarms +	ap-south-1a	–