

# Lab 5 – Terraform Variables

**Creator:** Sandeep Kumar Sharma

---

## Learning Objectives

- Understand what variables are in Terraform.
  - Learn why variables are used and their real-world importance.
  - Learn how to declare a variable inside `main.tf`.
  - Learn how to separate variables into a dedicated `variables.tf` file.
  - Learn how to assign values to variables.
- 

## Learning Outcome

By the end of this lab, the learner will be able to:

- Explain the purpose of variables in Terraform.
  - Declare and use variables inside the Terraform configuration.
  - Structure Terraform files cleanly by separating variables.
  - Deploy AWS resources using variables.
- 

## Concept Explanation (Natural Style)

Let's understand variables in Terraform using a simple, real-life example.

Imagine you're baking a cake. You use:

- 1 cup of sugar
- 2 cups of flour
- 1 spoon of vanilla

Now imagine tomorrow, you want to make a bigger cake. Will you rewrite the entire recipe again? No. You simply change the quantity.

Terraform works the same way.

### 👉 What is a variable in Terraform?

A variable is just a placeholder for a value. Instead of hardcoding values, we store them in variables.

**Example:** Instead of doing this:

```
instance_type = "t2.micro"
```

We write:

```
instance_type = var.instance_type
```

Now we can easily change the value in one place.

### 👉 Why do we use variables?

Because variables make your code:

- **Reusable** (use same code for Dev/QA/Prod)
- **Clean** (no hardcoded messy values)
- **Scalable** (large projects become easy to manage)
- **Configurable** (make changes without touching main logic)

### 👉 Role of variables in Terraform

Variables allow you to:

- change AWS region
- change instance size
- change bucket name
- change environment name without rewriting your entire Terraform code.

This is super helpful in real companies where you maintain hundreds of resources.

---

## 😡 Part 1: Declare Variables Inside main.tf (Simplest Approach)

This is the easiest form of using variables.

### Step 1: Create Lab Folder

```
mkdir terraform-lab5-variables  
cd terraform-lab5-variables
```

## Step 2: Create main.tf

```
touch main.tf
```

## Step 3: Write Terraform Code Using Variables

Open `main.tf` and add:

```
provider "aws" {
  region = var.aws_region
}

variable "aws_region" {
  description = "Region for AWS resources"
  type        = string
  default     = "ap-south-1"
}

variable "instance_type" {
  description = "EC2 instance size"
  type        = string
  default     = "t2.micro"
}

resource "aws_instance" "lab5_ec2" {
  ami           = "ami-052c08d70def0ac62" # Amazon Linux 2 (Mumbai)
  instance_type = var.instance_type

  tags = {
    Name = "Terraform-Lab5-EC2"
  }
}
```

## Explanation:

- We declared variables using the `variable` block.
- We assigned default values.
- We used variables inside the EC2 resource.

## Step 4: Run Terraform Commands

```
terraform init
terraform plan
terraform apply
```

Type **yes**.

EC2 instance will be created using variable values.

---

## Part 2: Using a Separate variables.tf File (Recommended Style)

In real projects, we never declare variables inside `main.tf`. We keep variable declarations in a dedicated file for clarity.

### Step 1: Split the Code

Keep `main.tf` clean.

#### main.tf

```
provider "aws" {
  region = var.aws_region
}

resource "aws_instance" "lab5_ec2" {
  ami          = "ami-052c08d70def0ac62"
  instance_type = var.instance_type

  tags = {
    Name = "Terraform-Lab5-EC2"
  }
}
```

#### variables.tf

Create a new file:

```
touch variables.tf
```

Add:

```
variable "aws_region" {
  description = "Region for AWS resources"
  type        = string
  default     = "ap-south-1"
```

```
}  
  
variable "instance_type" {  
  description = "EC2 instance size"  
  type        = string  
  default     = "t2.micro"  
}
```

## Step 2: Run Terraform Again

```
terraform plan  
terraform apply
```

Terraform will automatically detect variables from `variables.tf`.

---

## Summary

In this lab, you learned:

- What variables are in Terraform.
  - Why variables are important in real-world projects.
  - How to declare a variable inside `main.tf`.
  - How to separate variables into a dedicated `variables.tf` file.
  - How variables make your Terraform code cleaner and more scalable.
- 

**End of Lab 5**