

LAB-6

Terraform Multiple tfvars Files

Step 1: Create dev.tfvars and prod.tfvars

```
main.tf  ×  instance.tf  var.tf

main.tf > provider "aws" > secret_key
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "5.31.0"
6     }
7   }
8 }
9
10 provider "aws" {
11
12   region = "ap-south-1"
13   access_key = "AKIATJHVFEM70WRV3DM7"
14   secret_key = "0f6L+bKZ9nyf+nsVw9YIfN9AKcSyquaUuiPzmjPh"
15 }
```

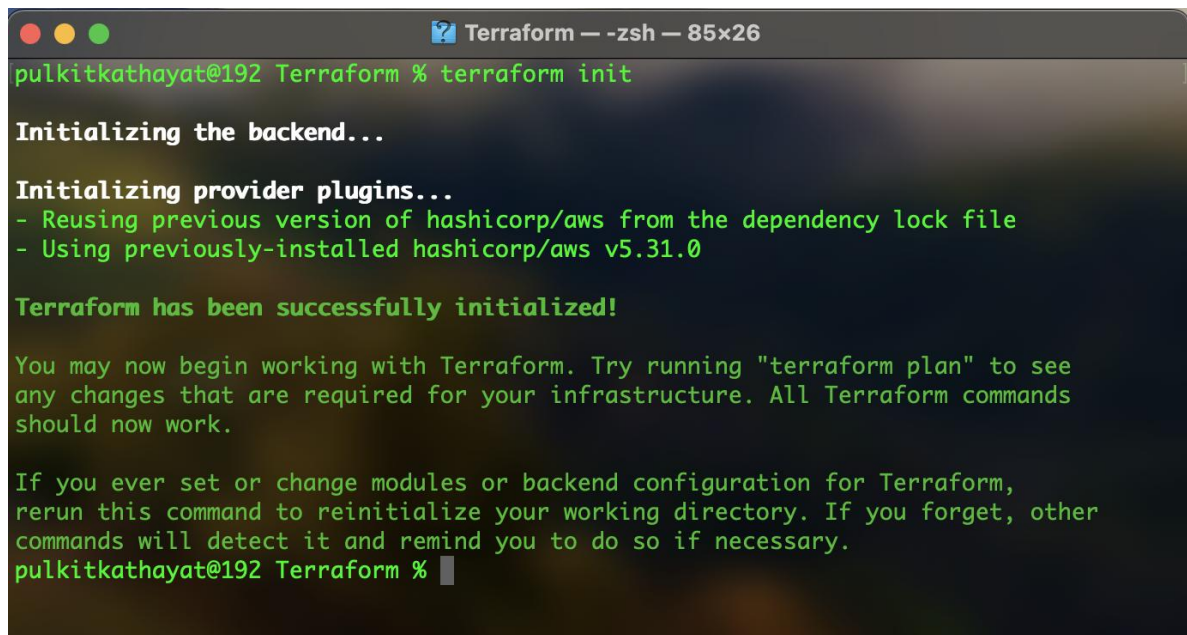
```
main.tf  instance.tf  ×  dev.tfvars  var.tf

instance.tf > resource "aws_instance" "lab6"
1 resource "aws_instance" "lab6" {
2   instance_type = var.instance_type
3   ami = var.ami_id
4   count = 1
5   tags = {
6     Name = "lab6-b3-2"
7   }
8
9 }
10
```

```
main.tf instance.tf dev.tfvars × var.tf
dev.tfvars > 
1 instance_type = "t2.micro"
2 ami_id = "ami-03f4878755434977f"
```

```
main.tf instance.tf dev.tfvars prod.tfvars ×
prod.tfvars > 
1 instance_type = "t2.micro"
2 ami_id = "ami-05a5bb48beb785bf1"
```

```
main.tf instance.tf dev.tfvars prod.tfvars var.tf ×
var.tf >  variable "instance_type"
1 variable "instance_type" {
2   type = string
3 }
4 variable "ami_id" {
5   type = string
6 }
```

A terminal window titled "Terraform — -zsh — 85x26" showing the output of the "terraform init" command. The output includes messages about initializing the backend and provider plugins, and a confirmation that Terraform has been successfully initialized. The prompt "pulkitkathayat@192 Terraform %" is visible at the bottom.

```
pulkitkathayat@192 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.
pulkitkathayat@192 Terraform %
```

Step 3: To run terraform plan we need to use `-var-` file=`dev.tfvars` or `-var- file=prod.tfvars`

```
pulkitkathayat@192 Terraform % terraform plan -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.lab6[0] will be created

```
+ resource "aws_instance" "lab6" {
  + 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" = "lab6-b3-2"
  }
  + tags_all                 = {
    + "Name" = "lab6-b3-2"
  }
  + 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.

```
pulkitkathayat@192 Terraform % terraform plan -var-file=prod.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.lab6[0] will be created
+ resource "aws_instance" "lab6" {
  + ami                        = "ami-05a5bb48beb785bf1"
  + 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" = "lab6-b3-2"
  }
  + tags_all                    = {
    + "Name" = "lab6-b3-2"
  }
  + 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.

Step 4: To run terraform apply and destroy we need to use `-var-file=dev.tfvars` or `-var-file=prod.tfvars`


```
pulkitkathayat@192 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.lab6[0] will be created
+ resource "aws_instance" "lab6" {
  + 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" = "lab6-b3-2"
  }
  + tags_all                   = {
    + "Name" = "lab6-b3-2"
  }
  + 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.

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.lab6[0]: Creating...

aws_instance.lab6[0]: Still creating... [10s elapsed]

aws_instance.lab6[0]: Still creating... [20s elapsed]

aws_instance.lab6[0]: Creation complete after 22s [id=i-020b50c406f4f07e3]

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

```
pulkitkathayat@192 Terraform % █
```

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:

Instances (1) info

Find Instance by attribute or tag (case-sensitive)

Any state

Launch instances

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Platform
<input type="checkbox"/>	lab6-b3-2	i-020b50c406f4f07e3	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1a	ec2

EC2 Dashboard

EC2 Global View

Events

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

```
pulkitkathayat@192 Terraform % terraform apply -var-file=prod.tfvars
aws_instance.lab6[0]: Refreshing state... [id=i-020b50c406f4f07e3]

Terraform used the selected providers to generate the following execution plan. Resource actions
are indicated with the following symbols:
-/+ destroy and then create replacement

Terraform will perform the following actions:

# aws_instance.lab6[0] must be replaced
-/+ resource "aws_instance" "lab6" {
  ~ ami                    = "ami-03f4878755434977f" -> "ami-05a5bb48beb785bf1" # forces replacement
  ~ arn                    = "arn:aws:ec2:ap-south-1:225999921982:instance/i-020b50c406f4f07e3" -> (known after apply)
  ~ associate_public_ip_address = true -> (known after apply)
  ~ availability_zone        = "ap-south-1a" -> (known after apply)
  ~ cpu_core_count           = 1 -> (known after apply)
  ~ cpu_threads_per_core     = 1 -> (known after apply)
  ~ disable_api_stop         = false -> (known after apply)
  ~ disable_api_termination  = false -> (known after apply)
  ~ ebs_optimized            = false -> (known after apply)
  ~ hibernation              = false -> null
  + host_id                  = (known after apply)
  + host_resource_group_arn  = (known after apply)
  + iam_instance_profile     = (known after apply)
  ~ id                       = "i-020b50c406f4f07e3" -> (known after apply)
  ~ instance_initiated_shutdown_behavior = "stop" -> (known after apply)
  + instance_lifecycle       = (known after apply)
  ~ instance_state           = "running" -> (known after apply)
  ~ ipv6_address_count       = 0 -> (known after apply)
  ~ ipv6_addresses           = [] -> (known after apply)
  + key_name                  = (known after apply)
  ~ monitoring               = false -> (known after apply)
  + outpost_arn              = (known after apply)
  + password_data            = (known after apply)
  + placement_group          = (known after apply)
  ~ placement_partition_number = 0 -> (known after apply)
  ~ primary_network_interface_id = "eni-0c29d8e249722e23e" -> (known after apply)
  ~ private_dns              = "ip-172-31-40-254.ap-south-1.compute.internal" -> (known after apply)
  ~ private_ip               = "172.31.40.254" -> (known after apply)
  ~ public_dns               = "ec2-3-110-159-40.ap-south-1.compute.amazonaws.com" -> (known after apply)
  ~ public_ip                = "3.110.159.40" -> (known after apply)
  ~ secondary_private_ips    = [] -> (known after apply)
  ~ security_groups          = [
    - "default",
  ] -> (known after apply)
  + spot_instance_request_id = (known after apply)
  ~ subnet_id               = "subnet-0d606c76cf74f48ab" -> (known after apply)
  ~ tags                    = {
    "Name" = "lab6-b3-2"
  }
  ~ tenancy                  = "default" -> (known after apply)
  + user_data                = (known after apply)
  + user_data_base64        = (known after apply)
}
```

```

- 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-0a419db11b4262e69" -> null
  - volume_size          = 8 -> null
  - volume_type          = "gp2" -> null
}
}

```

Plan: 1 to add, 0 to change, 1 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.lab6[0]: Destroying... [id=i-020b50c406f4f07e3]
aws_instance.lab6[0]: Still destroying... [id=i-020b50c406f4f07e3, 10s elapsed]
aws_instance.lab6[0]: Still destroying... [id=i-020b50c406f4f07e3, 20s elapsed]
aws_instance.lab6[0]: Still destroying... [id=i-020b50c406f4f07e3, 30s elapsed]
aws_instance.lab6[0]: Destruction complete after 30s
aws_instance.lab6[0]: Creating...
aws_instance.lab6[0]: Still creating... [10s elapsed]
aws_instance.lab6[0]: Still creating... [20s elapsed]
aws_instance.lab6[0]: Still creating... [30s elapsed]
aws_instance.lab6[0]: Creation complete after 32s [id=i-08338fcdb778b64b4]

```

Apply complete! Resources: 1 added, 0 changed, 1 destroyed.
pulkitkathayat@192 Terraform %

The screenshot shows the AWS Management Console for the 'Pulkit Singh Kathayat' account in the 'Mumbai' region. The 'Instances (2)' page is active, showing a list of EC2 instances. The first instance, 'lab6-b3-2' (ID: i-020b50c406f4f07e3), is in a 'Terminated' state. The second instance, also named 'lab6-b3-2' (ID: i-08338fcdb778b64b4), is in a 'Running' state. The console interface includes a sidebar with navigation links, a top navigation bar with the AWS logo and account information, and a main content area with filters and a table of instances.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Platform
lab6-b3-2	i-020b50c406f4f07e3	Terminated	t2.micro	-	View alarms +	ap-south-1a	-
lab6-b3-2	i-08338fcdb778b64b4	Running	t2.micro	Initializing	View alarms +	ap-south-1a	ec2


```
pulkitkathayat@192 Terraform % terraform destroy -var-file=prod.tfvars
aws_instance.lab6[0]: Refreshing state... [id=i-08338fcdb778b64b4]
```

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.lab6[0] will be **destroyed**

```
- resource "aws_instance" "lab6" {
  - ami                               = "ami-05a5bb48beb785bf1" -> null
  - arn                               = "arn:aws:ec2:ap-south-1:225999921982:instance/i-08338fcdb778b64b4" -> 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-08338fcdb778b64b4" -> 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-09b6a9a11308a296a" -> null
  - private_dns                       = "ip-172-31-44-133.ap-south-1.compute.internal" -> null
  - private_ip                        = "172.31.44.133" -> null
  - public_dns                         = "ec2-13-233-86-62.ap-south-1.compute.amazonaws.com" -> null
  - public_ip                          = "13.233.86.62" -> null
  - secondary_private_ips               = [] -> null
  - security_groups                    = [
    - "default",
  ] -> null
  - source_dest_check                  = true -> null
  - subnet_id                          = "subnet-0d606c76cf74f48ab" -> null
  - tags                              = {
    - "Name" = "lab6-b3-2"
  } -> null
  - tags_all                           = {
    - "Name" = "lab6-b3-2"
  } -> null
  - tenancy                            = "default" -> 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                  = 3000 -> null
  - tags                  = {} -> null
  - throughput           = 125 -> null
  - volume_id            = "vol-0b5fc1e1565ffe8dd" -> null
  - volume_size          = 10 -> null
  - volume_type          = "gp3" -> 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.lab6[0]: Destroying... [id=i-08338fcd778b64b4]
aws_instance.lab6[0]: Still destroying... [id=i-08338fcd778b64b4, 10s elapsed]
aws_instance.lab6[0]: Still destroying... [id=i-08338fcd778b64b4, 20s elapsed]
aws_instance.lab6[0]: Still destroying... [id=i-08338fcd778b64b4, 30s elapsed]
aws_instance.lab6[0]: Destruction complete after 30s

```

Destroy complete! Resources: 1 destroyed.
pulkithathayat@192 Terraform %

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	lab6-b3-2	i-020b50c406f4f07e3	Terminated	t2.micro	-	View alarms	ap-south-1a
<input type="checkbox"/>	lab6-b3-2	i-08338fcd778b64b4	Terminated	t2.micro	-	View alarms	ap-south-1a

**When we run terraform apply --var-file=prod.tfvars
previously created terraform apply --var-file=dev.tfvars
automatically destroy.**

