Lab 14: Terraform Workspaces – Environment Management Using Workspaces

Author: Dr. Sandeep Kumar Sharma
Level: Beginner to Intermediate
Platform: Ubuntu Linux + Microsoft Azure
Prerequisite: Lab 1 to Lab 13

Learning Objective

Participants will learn:

- What Terraform workspace is
- Why Terraform workspace is needed
- How Terraform workspace works
- How to create a workspace
- How to switch between workspaces
- How workspaces manage state

Learning Outcome

After completing this lab, participants will:

- Understand multi-environment management
- Use Terraform workspaces
- Manage separate state files
- Build environment isolation

Concept Explanation

What is Terraform Workspace?

A Terraform workspace is a **separate working environment**.

Each workspace has: - Its own state file - Its own infrastructure tracking - Its own resource mapping

Default workspace name:

```
default
```

Why We Need Terraform Workspace?

Terraform workspace is used to:

- Create multiple environments
- Separate state files
- Isolate infrastructure
- Avoid resource conflicts
- Manage Dev / Test / Prod
- Enable multi-environment deployments

How Terraform Workspace Works

Each workspace has:

```
terraform.tfstate
```

But internally Terraform stores them as:

```
terraform.tfstate.d/<workspace_name>/terraform.tfstate
```

Hands-On Lab

Step 1: Create New Folder

```
mkdir terraform-workspace-lab
cd terraform-workspace-lab
```

Step 2: Create Terraform File

```
touch main.tf
nano main.tf
```

Add:

```
provider "azurerm" {
  features {}
}

resource "azurerm_resource_group" "rg" {
  name     = "rg-workspace-${terraform.workspace}"
  location = "East US"
}
```

Step 3: Initialize Terraform

```
terraform init
```

Workspace Operations

Step 4: List Workspaces

```
terraform workspace list
```

Step 5: Create New Workspace

```
terraform workspace new dev
```

Step 6: Switch Workspace

```
terraform workspace select dev
```

Step 7: Apply in dev Workspace

```
terraform apply
```

Type:

```
yes
```

Resource Group created:

```
rg-workspace-dev
```

Step 8: Create prod Workspace

```
terraform workspace new prod
```

```
terraform workspace select prod
```

Step 9: Apply in prod Workspace

```
terraform apply
```

Type:

```
yes
```

Resource Group created:

```
rg-workspace-prod
```

Verification

Azure Portal:

- rg-workspace-dev
- rg-workspace-prod

Understanding State Isolation

Each workspace has:

- Separate state
- Separate resource tracking
- Independent lifecycle

Workspace Commands Summary

```
terraform workspace list
terraform workspace new <name>
terraform workspace select <name>
terraform workspace show
```

Cleanup

Select each workspace and destroy:

```
terraform workspace select dev
terraform destroy
```

```
terraform workspace select prod
terraform destroy
```