**Extra Lab Exercise-2 - Terraform Workspace Management**

**Objective:**

Learn to create, switch, and manage Terraform workspaces to deploy isolated environments (e.g., dev and prod) using a single configuration.

**1. Setup: Project Directory**

mkdir tf-workspace-lab

cd tf-workspace-lab

**2. Create Terraform Configuration File (main.tf)**

provider "aws" {

region = "us-east-1"

}

resource "aws\_s3\_bucket" "workspace\_bucket" {

bucket = "workspace-lab-bucket-${terraform.workspace}"

acl = "private"

tags = {

Environment = terraform.workspace

}

}

terraform.workspace is a built-in variable that returns the current workspace name. It helps dynamically name resources per workspace.

**3. Initialize Terraform**

terraform init

**4. View Existing Workspaces**

terraform workspace list

Output will show:

\* default

**5. Create a New Workspace (dev)**

terraform workspace new dev

Check again:

terraform workspace list

You will see:

default

\* dev

**6. Apply Configuration in dev**

terraform apply -auto-approve

An S3 bucket will be created with the name like:  
workspace-lab-bucket-dev

**7. Create and Switch to prod Workspace**

terraform workspace new prod

terraform apply -auto-approve

Creates: workspace-lab-bucket-prod

Now both dev and prod buckets are managed in isolated states!

**8. Switch Between Workspaces**

terraform workspace select dev

terraform show # Shows resources in 'dev'

terraform workspace select prod

terraform show # Shows resources in 'prod'

**9. Cleanup**

terraform workspace select dev

terraform destroy -auto-approve

terraform workspace select prod

terraform destroy -auto-approve

Finally, switch to default and delete extra workspaces:

terraform workspace select default

terraform workspace delete dev

terraform workspace delete prod