Terraform Workspaces: Overview & Best Practices

# What is a Terraform Workspace?

A Terraform workspace is an isolated instance of state data. Workspaces allow you to manage multiple versions of your infrastructure using the same configuration files but different state files. Each workspace maintains its own state, making it ideal for separating environments such as development, staging, and production.

# Why Are Workspaces Required?

Workspaces are useful when you want to use the same infrastructure code across multiple environments without maintaining separate copies of configuration files. They help avoid duplication and reduce complexity by reusing modules and .tf files while maintaining isolated states.

## Benefits of Using Workspaces

• Environment Isolation: Separate state for dev, staging, and production.  
• Code Reuse: Use the same Terraform code across environments.  
• Cleaner Structure: No need for multiple root directories.  
• Easier CI/CD Integration: Manage environments more cleanly.

# Example Usage of Terraform Workspaces

1. Create a new workspace:

terraform workspace new dev

2. Select an existing workspace:

terraform workspace select staging

3. List all workspaces:

terraform workspace list

4. Apply with environment-specific variables:

terraform apply -var-file=dev.tfvars

# Example Project Structure

terraform-project/  
├── main.tf  
├── variables.tf  
├── outputs.tf  
├── dev.tfvars  
├── staging.tfvars  
└── modules/  
 └── compute\_instance/  
 ├── main.tf  
 ├── variables.tf  
 └── outputs.tf