**“3\_Terraform Basics Demos for Providers And Terraform Initialization - Made with Clipchamp”**

**📚 Demo Outline: Terraform Basics (Providers & Initialization)**

**Demo 1: Terraform Initialization**

1. Write a minimal main.tf with only the AWS provider.
2. terraform {
3. required\_providers {
4. aws = {
5. source = "hashicorp/aws"
6. version = "6.11.0"
7. }
8. }
9. }
10. provider "aws" {
11. region = "us-east-1"
12. }
13. Run:
14. terraform init
15. Show .terraform/ folder + .terraform.lock.hcl file created.
16. Explain why lock files matter.

**Demo 2: Multiple Providers (AWS + Azure)**

1. Update main.tf to include **Azure provider** alongside AWS:
2. terraform {
3. required\_providers {
4. aws = {
5. source = "hashicorp/aws"
6. version = "6.11.0"
7. }
8. azurerm = {
9. source = "hashicorp/azurerm"
10. version = "4.7.0"
11. }
12. }
13. }
14. provider "aws" {
15. region = "us-east-1"
16. }
17. provider "azurerm" {
18. features {}
19. }
20. Run:
21. terraform init
22. Show how Terraform downloads **both providers**.

**Demo 3: Minimal Resources with Each Provider**

*(Safe, non-expensive resources)*

* Add AWS **S3 bucket**.
* Add Azure **Resource Group**.  
  Example:

resource "aws\_s3\_bucket" "demo\_bucket" {

bucket = "my-tf-demo-bucket-123456"

}

resource "azurerm\_resource\_group" "demo\_rg" {

name = "tf-demo-rg"

location = "East US"

}

1. Run:
2. terraform plan
3. terraform apply
4. Show creation in **AWS Console (S3)** and **Azure Portal (Resource Groups)**.
5. Clean up with:
6. terraform destroy

⚡ By structuring your Clipchamp video in these **three demos**, you’ll cover:

* Terraform initialization
* Multiple providers
* Resource provisioning basics