**🛠 Terraform Tutorial: Change & Apply with Regenerated Plan**

**Step 0: Setup Folder**

mkdir C:\terraform-demo

cd C:\terraform-demo

**Step 1: Create Initial main.tf**

terraform {

required\_providers {

local = {

source = "hashicorp/local"

version = "~> 2.0"

}

}

}

provider "local" {}

resource "local\_file" "hello" {

content = "Hello from PwSkills"

filename = "${path.module}/hello.txt"

}

**Step 2: Initialize Terraform**

terraform init

✅ Downloads local provider and sets up folder.

**Step 3: Apply Initial Configuration**

terraform apply -auto-approve

✅ Result:

* File hello.txt created:

Hello from PwSkills

* Terraform creates terraform.tfstate.

**Step 4: Save a Plan**

terraform plan -out=before-change2.tfplan

* Saves a **plan file** for potential future use.

**Step 5: Apply Saved Plan**

terraform apply before-change2.tfplan

✅ Result:

* File content remains:

Hello from PwSkills

**Step 6: Edit main.tf to New Content**

resource "local\_file" "hello" {

content = "Hello from PwSkills2"

filename = "${path.module}/hello.txt"

}

**Step 7: Regenerate the Plan**

****

terraform plan -out=before-change3.tfplan

* Terraform compares **new config** vs **current state**.
* Output shows:

~ resource "local\_file" "hello" {

content = "Hello from PwSkills" -> "Hello from PwSkills2"

}

**Step 8: Apply the Regenerated Plan**

terraform apply before-change3.tfplan

✅ Result:

* hello.txt updated:

Hello from PwSkills2

* State file updated accordingly.
* No new resource created, only in-place update.

**Step 9: Verify State**

terraform state list

terraform show

* Confirms local\_file.hello is tracked.
* Content in state = "Hello from PwSkills2"

**Step 10: Optional: Revert Again**

1. Edit main.tf back:

content = "Hello from PwSkills"

1. Regenerate plan:

terraform plan -out=revert.tfplan

1. Apply:

terraform apply revert.tfplan

✅ File content reverts safely.

**✅ Key Concepts Practiced**

1. Terraform **tracks resource state** in terraform.tfstate.
2. **Old plans become stale** if state/config changes.
3. **Option 1: Regenerate plan** ensures safe apply after edits.
4. Changes are applied **incrementally**, no new resources unless needed.