Terraform Task

**Name:Shaik Khaja Basha**

**Batch : Batch 11**

**Date : 18.07.2025**

**Task : File creations in .tf & make corrections**

1. **make a mistake in configuration->run validate->correct and execute for that i had like abc.txt file in terraform**

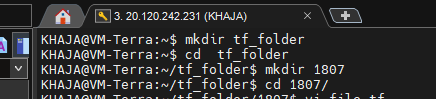
**Ans: Open your terminal and run:**

mkdir tf\_folder # Main folder for Terraform work

cd tf\_folder # Move into that folder

mkdir 1807 # Create a sub-folder for today's date or task

cd 1807 # Move into that sub-folder



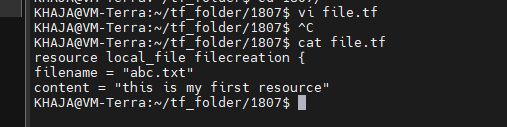
* Create Terraform configuration file
* vi file.tf
* In vi, press i to enter **insert mode**, then **paste this content**:

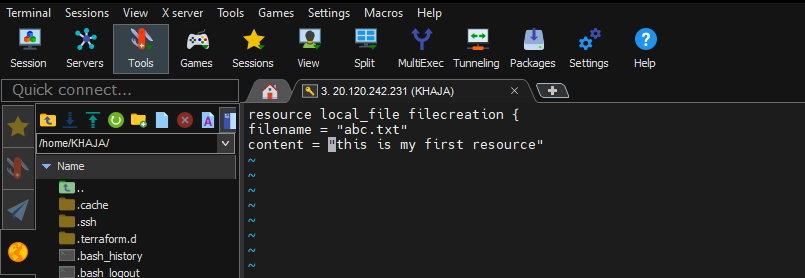
**Syntax** : resource "local\_file" "filecreation" {

filename = "abc.txt"

content = "this is my first resource"

}

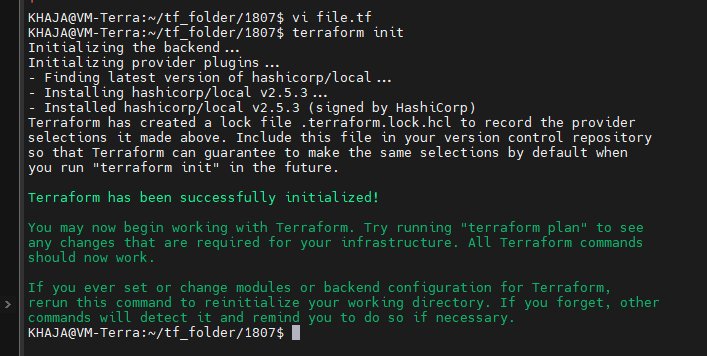




* Press Esc
* Type :wq and hit **Enter** to save and exit
* **Initialize Terraform**
* terraform init

This will download the required local provider plugin and set up .terraform directory.

**Output will look like:**

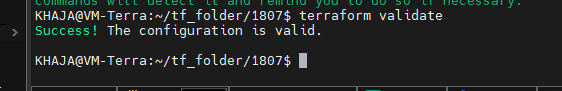


* **Validate the configuration:**

Check if your configuration is correct:

* terraform validate

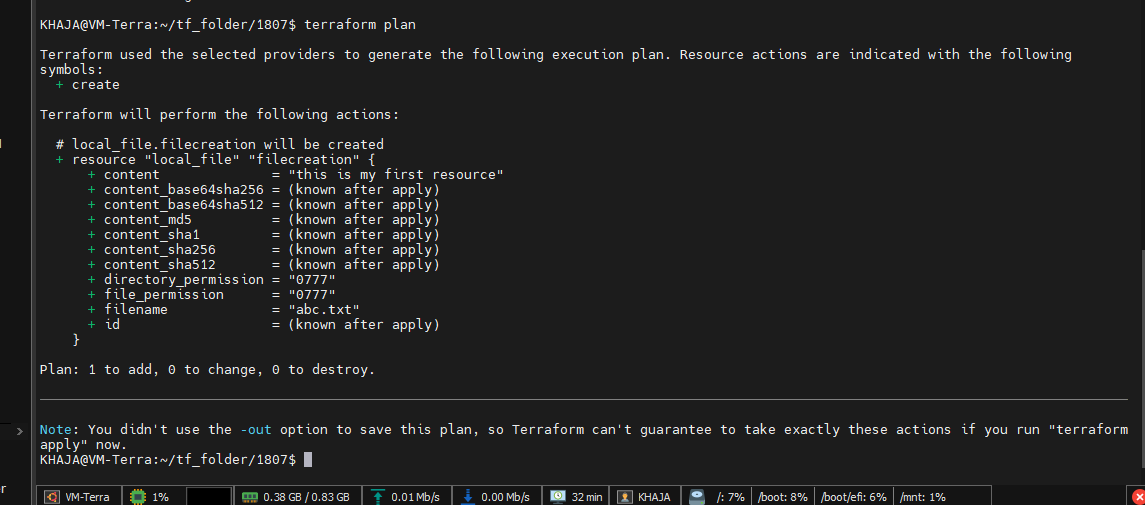
output:



Its show me as validate “Success”

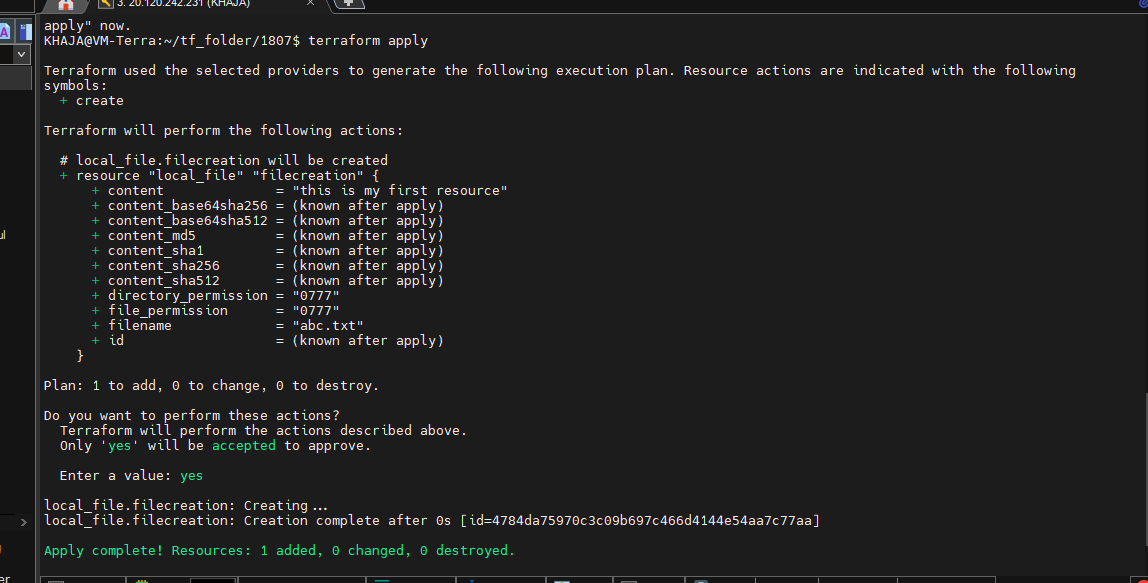
* **Plan the Infrastructure**
* terraform plan

This step shows what Terraform **will do** without actually doing it.

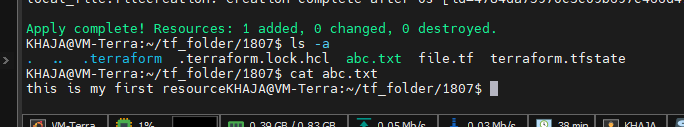


* **Apply the Configuration**
* terraform apply

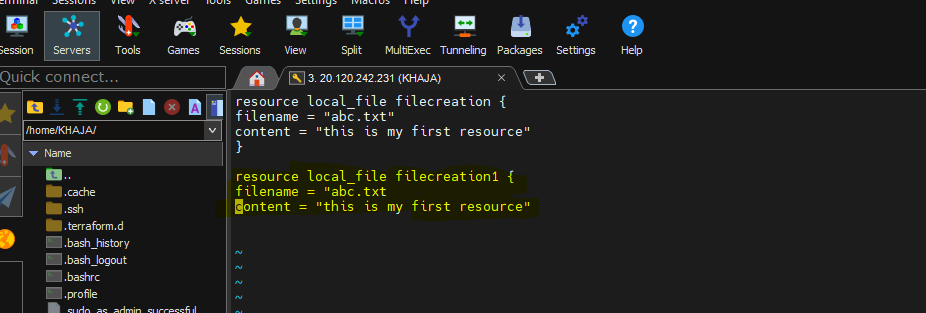
This step actually **creates the file**.



Verify Output

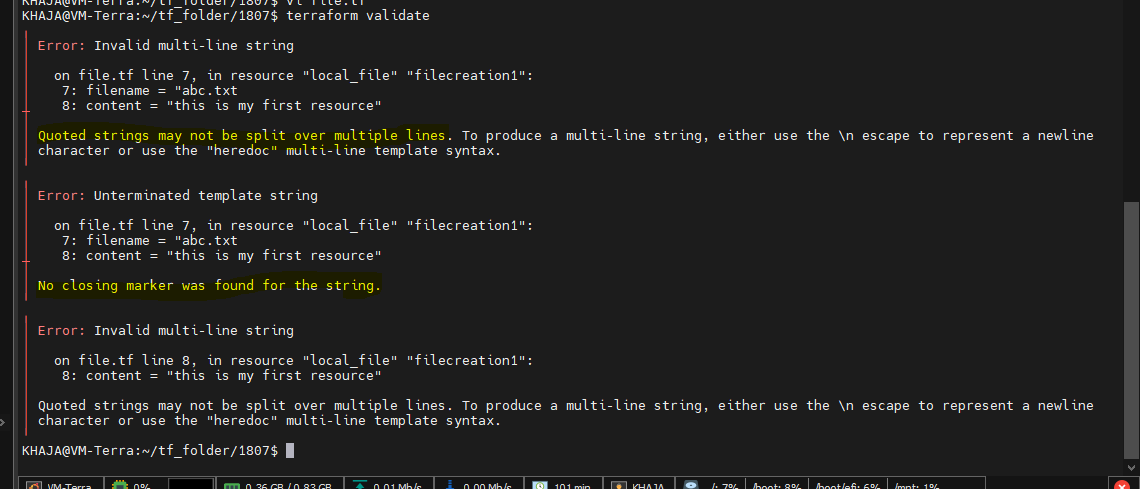


Add Terraform code with a mistake (Missing closing brace)



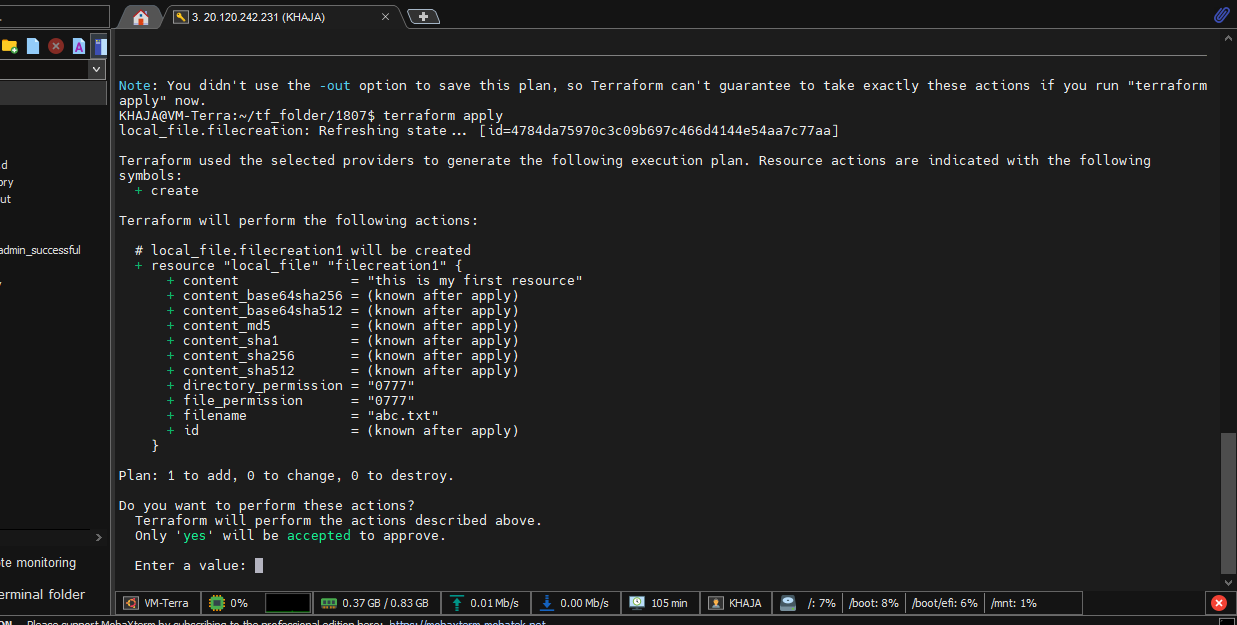
When I create a another resource within the same file but it show the error

* Terraform validate

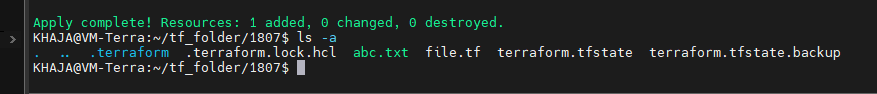


Lets correct the configuration and make execute

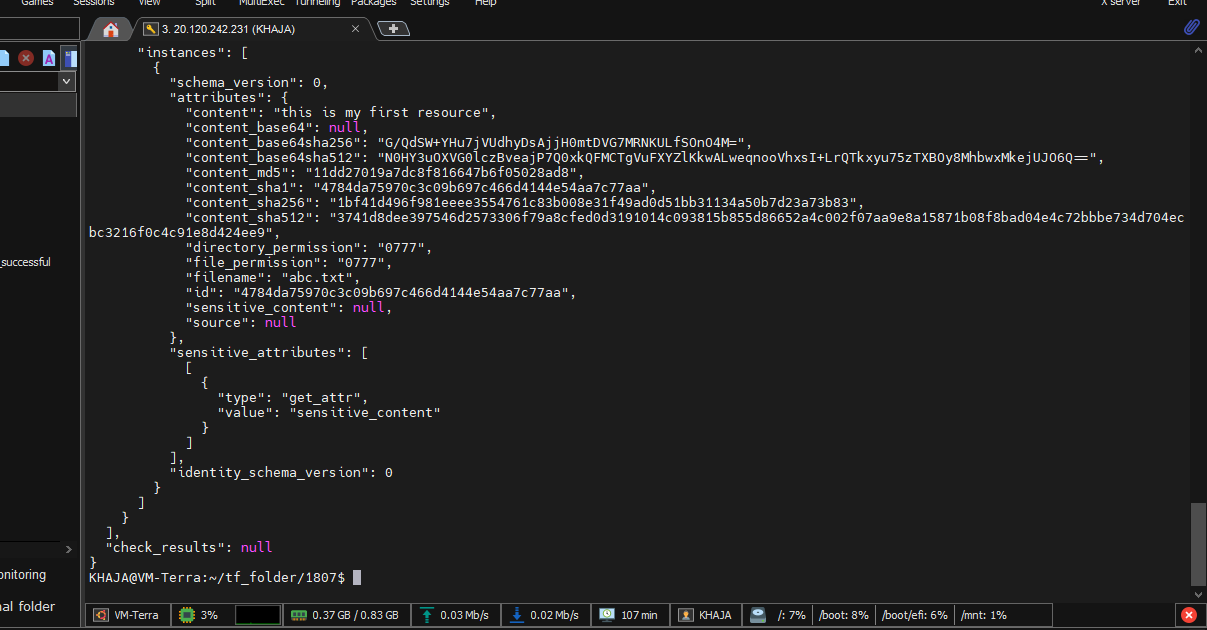
* Terraform apply



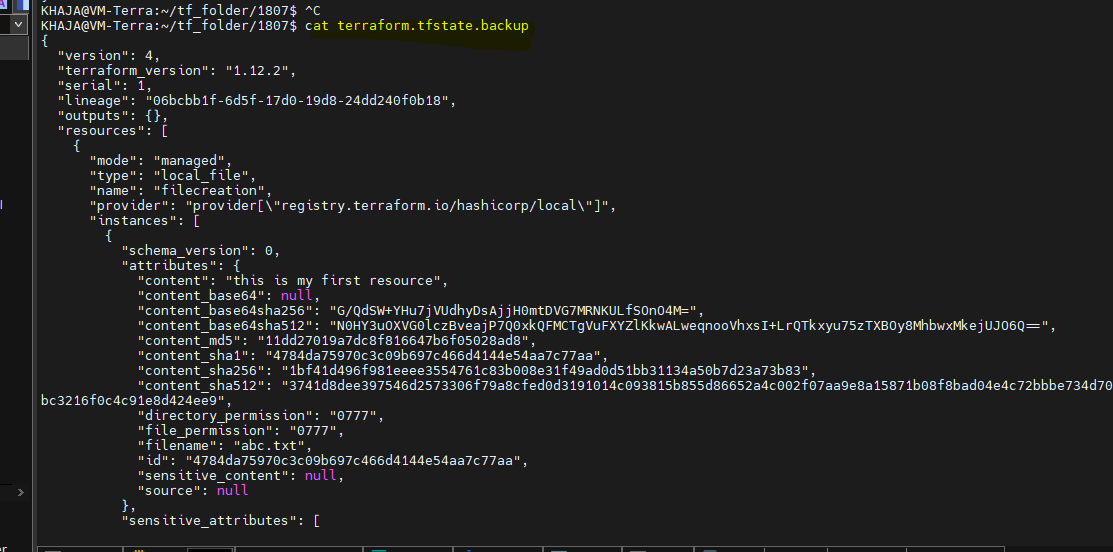
Know it will create a file with no error



When we check the state file it will show the previous execution



For the backup file it contain the previous execution



1. **create 3 .tf file with single resources inside**

Ans: three Terraform (.tf) files, each containing a single resource:

Creating files

Create a file name file1.tf using vi file1.tf

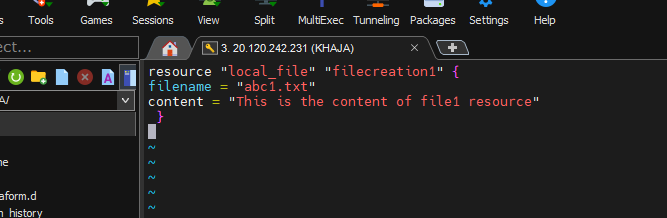
Enter the resource in file1.tf and save it esc:wq

resource "local\_file" "filecreation1" {

filename = "abc1.txt"

content = "This is the content of file1 resource"

}



Know we have created file1.tf

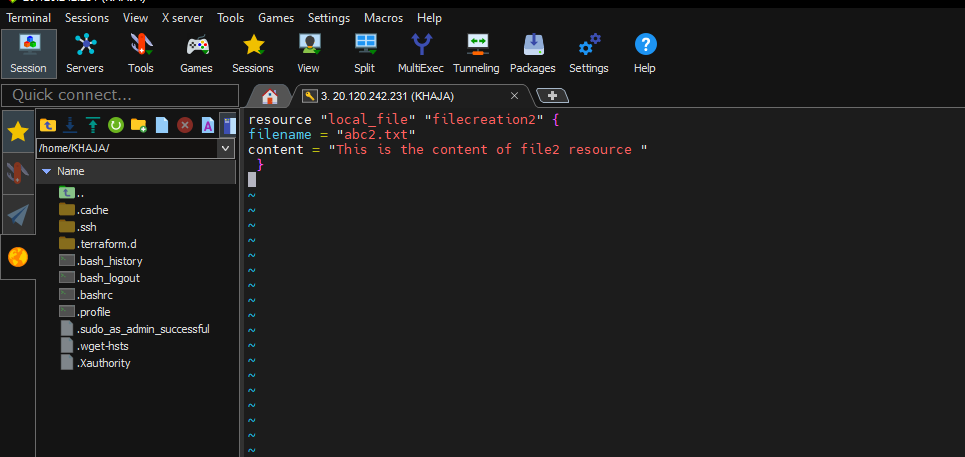
Create the second file content file2.tf and save it esc:wq

resource "local\_file" "filecreation2" {

filename = "abc2.txt"

content = "This is the content of file2 resource "

}



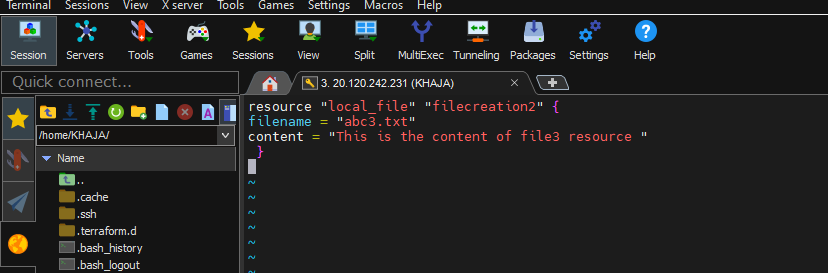
Create the third file content file3.tf and save it esc:wq

resource "local\_file" "filecreation2" {

filename = "abc3.txt"

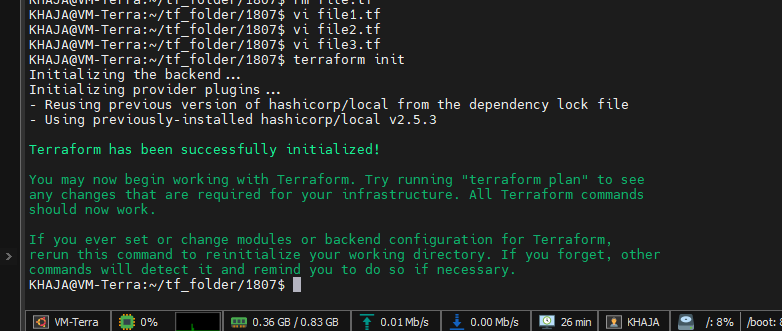
content = "This is the content of file3 resource "

}



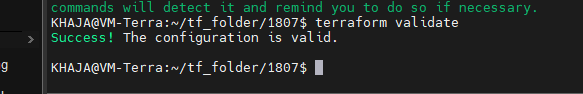
Run the command to Initialize Terraform

* terraform init



Will check for the validation

* terraform validate



After that we need to run the following commands

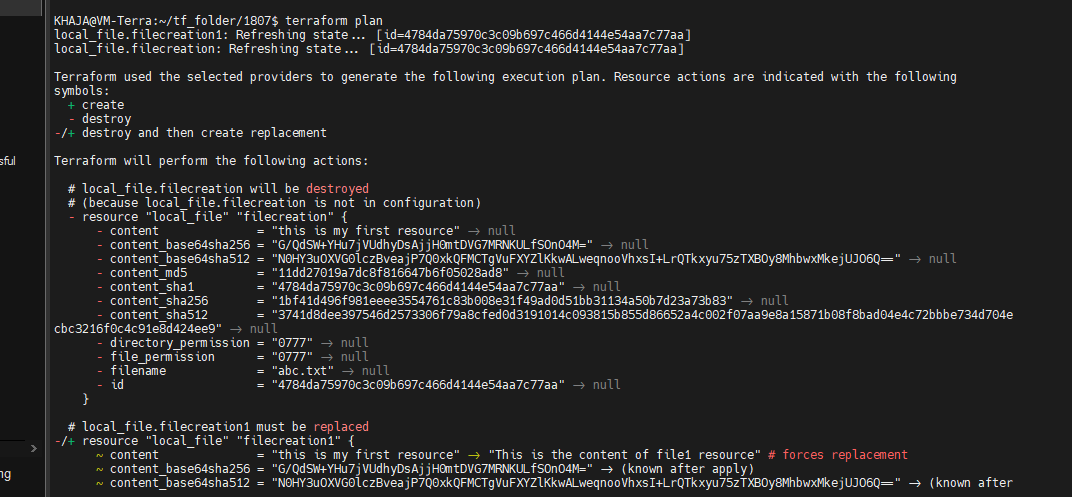
terraform plan

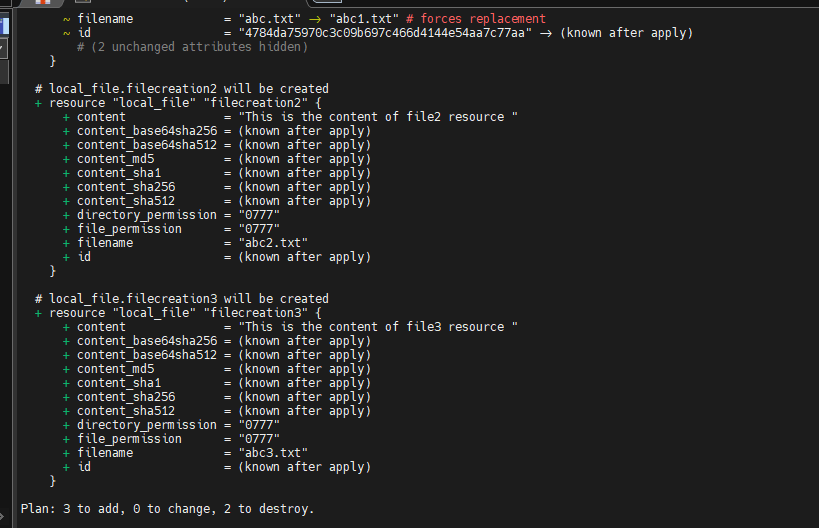
* it will show a preview or dry run.
* It shows like What Terraform wants to create, change, or delete.
* No changes are made to our real environment when running plan.

terraform apply

* This is the action step.
* Terraform uses our .tf files to actually: to Create, modify, or delete real resources (VMs, networks, etc.).
* We will usually get a confirmation prompt before changes happen like for proceeding yes/No

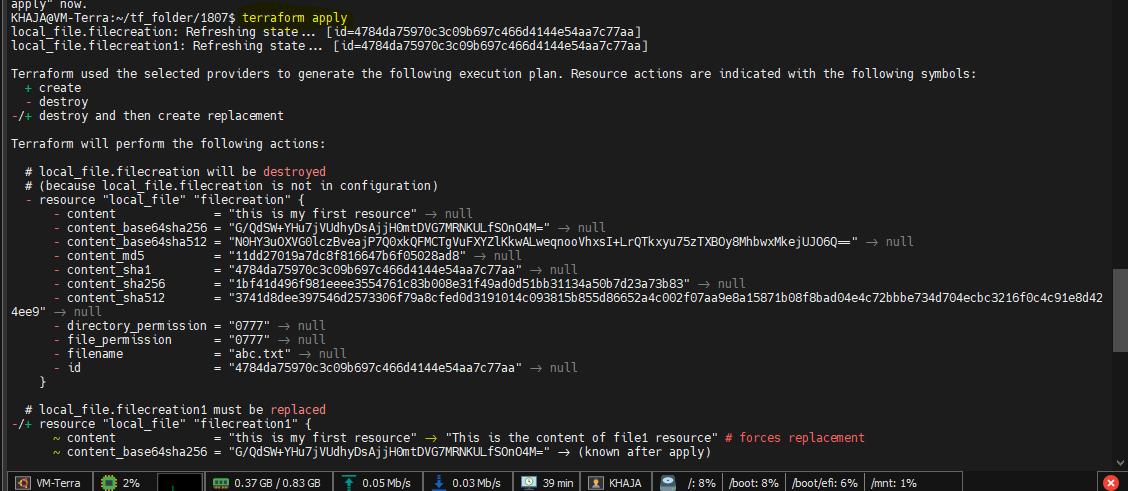
Run the command terraform plan

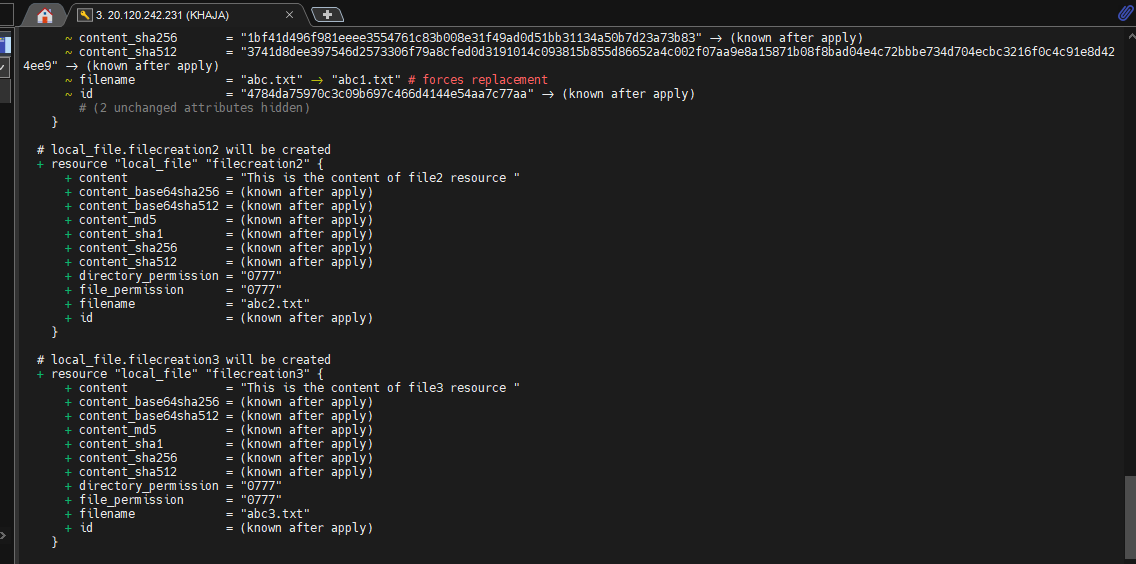


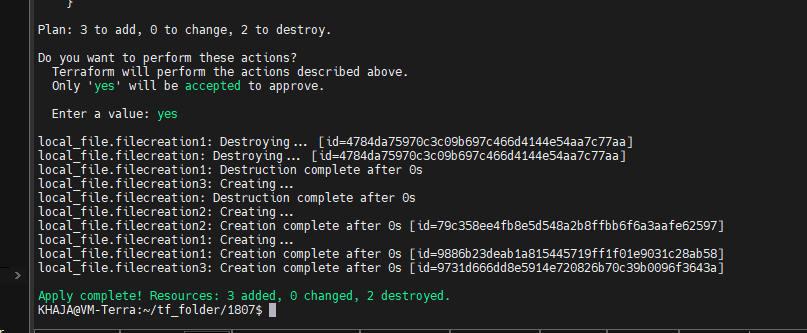


Run the command terraform apply

Terraform will show actions and prompt you for confirmation. Type yes when prompted.







1. **Create single .tf file with three resources in it**

**Ans:** We need to create a .tf files, with a three resources in it, here is a clear structure and example content for each file, using the [local\_file resource from the local provider]

**Create a single .tf file with:**

* **1 provider block**
* **3 resources:**
  + **abc1.txt**
  + **abc2.txt**
  + **abc3.txt**

**Creating file**

We create a file named main1.tf using vi file.tf

Enter the contents in file.tf as follows and save it using esc:wq

**resource "local\_file" "filecreation1" {**

**filename = "file\_1.txt"**

**content = "This is the content of file\_1."**

**}**

**resource "local\_file" "filecreation2" {**

**filename = "file\_2.txt"**

**content = "This is the content of file 2."**

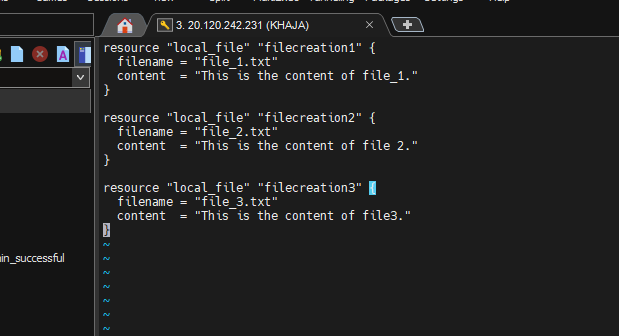
**}**

**resource "local\_file" "filecreation3" {**

**filename = "file\_3.txt"**

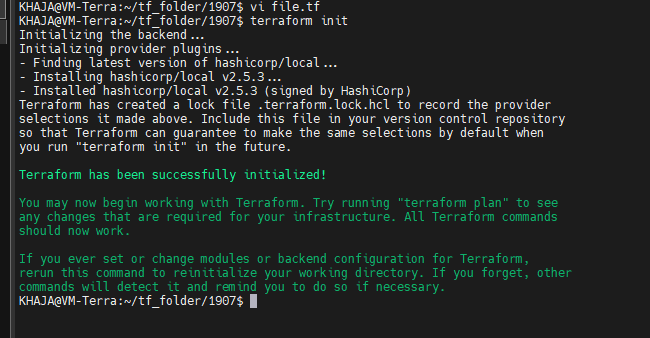
**content = "This is the content of file3."**

**}**

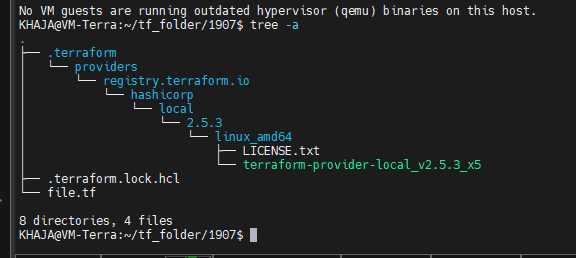


Initialize Terraform using command

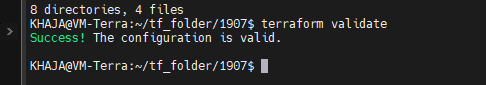
* terraform init



Check the contents using tree -a



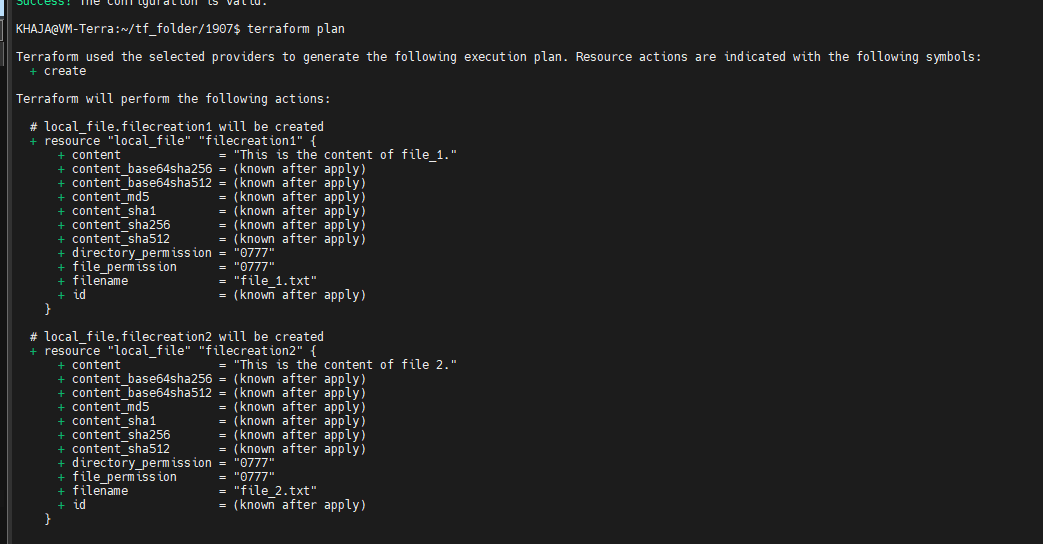
Run the command terraform validate

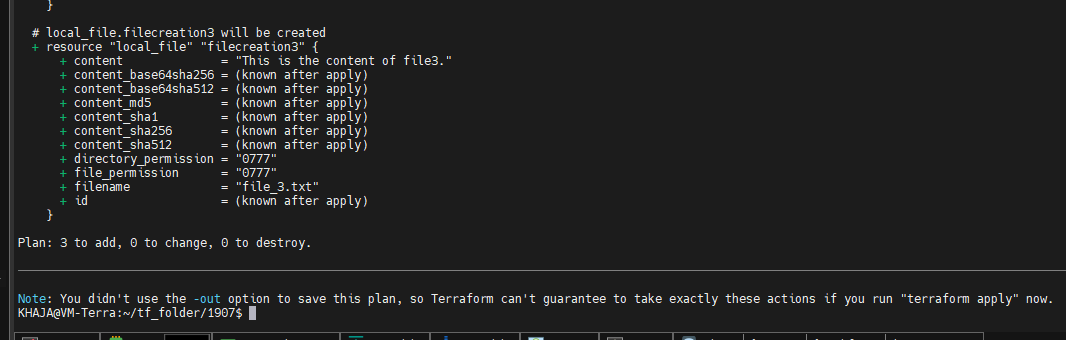


terraform plan

* it will show a preview or dry run.
* It shows like What Terraform wants to create, change, or delete.
* No changes are made to our real environment when running plan.

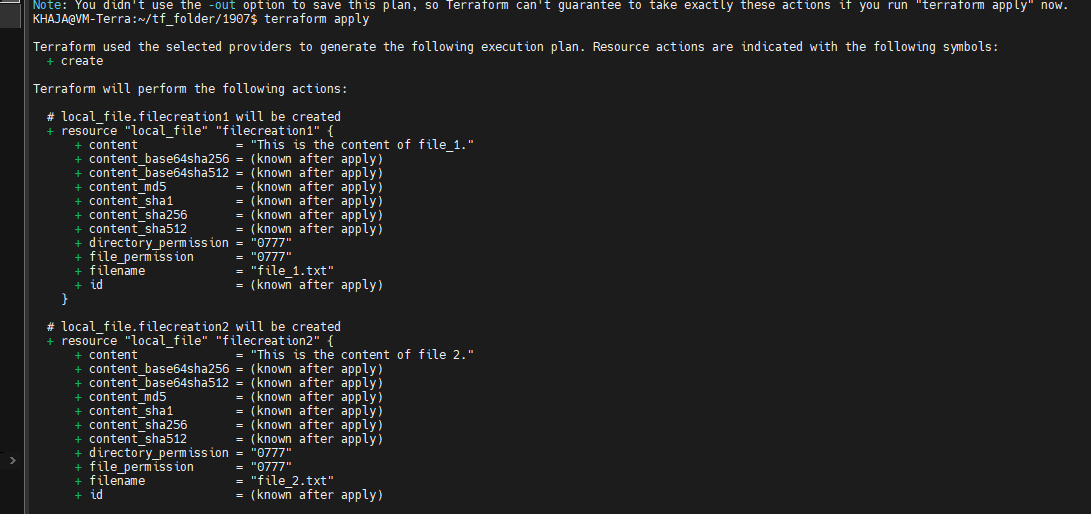
Run the command terraform plan

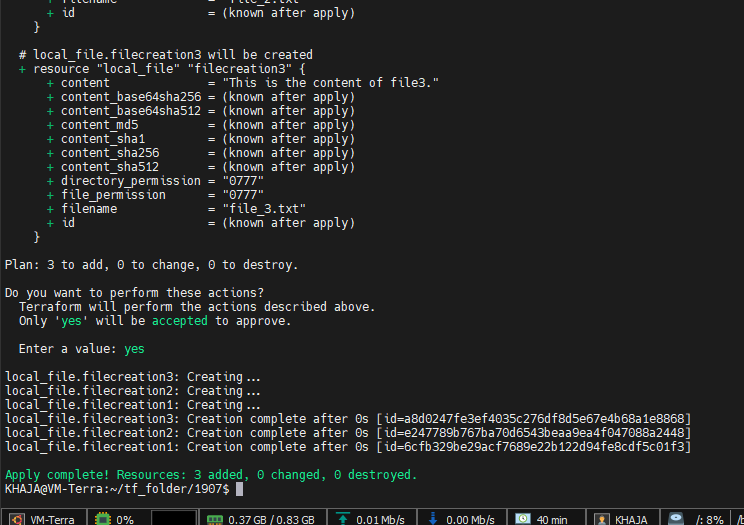




terraform apply

* This is the action step.
* Terraform uses our .tf files to actually: to Create, modify, or delete real resources (VMs, networks, etc.).
* We will usually get a confirmation prompt before changes happen like for proceeding yes/No





Check the contents of directory using tree -a

