



**Laboratorio**  
PRACTICA  
TERRAFORM



## CONTROL DE VERSIONES

<b>Elaborado por:</b> Jonatan Stiven Gutierrez	<b>No. de Versión:</b> 1.0.0
<b>Revisado por:</b>	<b>Fecha de revisión:</b>
<b>Aprobado por:</b>	<b>Fecha de Aprobación:</b>

### Historia de Modificaciones

No. de Versión	Fecha de Versión	Autor	Revisado por	Aprobado por	Descripción
1.0.0	21/02/2024	Jonatan Stiven Gutierrez			Documento Original

### Lista de distribución

Para	Acción*	Empresa	Firma/Medio de Entrega

\* Tipos de acción: Aprobar, Revisar, Informar, Archivar, Complementar, Asistir a junta, Otras (por favor especificar)

**Este documento fue elaborado por SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.**



## Contenido

<b>INTRODUCCION .....</b>	<b>4</b>
<b>PRERREQUISITOS .....</b>	<b>4</b>
<b>EJERCICIO 4:.....</b>	<b>5</b>



## INTRODUCCION

El siguiente documento proporciona una introducción detallada a los ejercicios realizados en el laboratorio.

## PRERREQUISITOS

- Tener entendimiento de los temas vistos anteriormente.



## EJERCICIO 4:

En este ejercicio vamos a copiar la carpeta del ejercicio 1, manejaremos terraform fmt y terraform validate.

- `cp -a practica_01 practica_04`

```
jonatan@ubuntu:~$ cd practica_terraform/  
jonatan@ubuntu:~/practica_terraform$ cp -a practica_01 practica_04  
jonatan@ubuntu:~/practica_terraform$ ls -ltr  
total 20  
drwxrwxr-x 3 jonatan jonatan 4096 feb 22 10:21 practica_04  
drwxrwxr-x 3 jonatan jonatan 4096 feb 22 10:21 practica_01  
drwxrwxr-x 3 jonatan jonatan 4096 feb 22 22:26 practica_02  
drwxrwxr-x 3 jonatan jonatan 4096 feb 22 23:50 practica_02.1  
drwxrwxr-x 3 jonatan jonatan 4096 feb 23 15:28 practica_03  
jonatan@ubuntu:~/practica_terraform$
```

1. Ingresamos a la nueva carpeta, hacemos un `ls -ltr`, y un `cat` al archivo "terraform.tf".

```
jonatan@ubuntu:~/practica_terraform$ cd practica_04  
jonatan@ubuntu:~/practica_terraform/practica_04$ ls -ltr  
total 12  
-rw-rw-r-- 1 jonatan jonatan 142 feb 22 09:30 terraform.tf  
-rw-rw-r-- 1 jonatan jonatan 1472 feb 22 10:21 terraform.tfstate.backup  
-rw-rw-r-- 1 jonatan jonatan 180 feb 22 10:21 terraform.tfstate  
jonatan@ubuntu:~/practica_terraform/practica_04$ cat terraform.tf  
resource "local_file" "cali" {  
    content = "Hola, grupo devops 2024, este es un ejemplo de terraform"  
    filename = "seti.txt"  
}  
  
jonatan@ubuntu:~/practica_terraform/practica_04$
```

2. Como vemos ese archivo muestra nuestro primero código de terraform, para este ejemplo vamos a corregir si un código está mal indentado.
  - Se llama indentación de código al hecho de utilizar sangrado (mover ligeramente hacia la derecha) las líneas de código, facilitando así la lectura, e indicando visualmente si nos encontramos en el interior de una estructura concreta, en el programa principal, etc...



3. Ejecutamos el siguiente comando para editar el archivo desde consola.
  - `sudo nano terraform.tf`
  - Una vez escrito el anterior comando nos mostrara lo siguiente:

```
GNU nano 6.2 terraform.tf
resource "local_file" "cali" {
  content = "Hola, grupo devops 2024, este es un ejemplo de terraform"
  filename = "seti.txt"
}
```

4. Editamos el archivo para que nos quede así.
  - Para moverse entre líneas, usar las flechas del teclado.

```
GNU nano 6.2
resource "local_file" "cali" {
content = "Hola, grupo devops 2024, este es un ejemplo de terraform"
filename = "seti.txt"
}
```

5. Ahora vamos a guardar los cambios con los siguientes comandos.
  - Control + o, para guardar.

```
Nombre del archivo a escribir [Formato DOS]: terraform.tf
Ayuda Formato DOS Añadir Respalda fichero
Cancelar Formato Mac Anteponer Navegar
```



- Damos enter para seguir con el mismo nombre del archivo.
- Ahora vamos a salir del editor nano, para ello presionamos "Control + x"
- Una vez salido del editor, hacemos el comando cat sobre el archivo.

```
jonatan@ubuntu:~/practica_terraform/practica_04$ cat terraform.tf
resource "local_file" "cali" {
  content = "Hola, grupo devops 2024, este es un ejemplo de terraform"
  filename = "seti.txt"
}
jonatan@ubuntu:~/practica_terraform/practica_04$
```

6. Ahora vamos a ejecutar el comando "terraform fmt", lo que hace este comando es arreglar la indentación del archivo.
- "terraform fmt", Hace un arreglo a nivel general en la carpeta en la que se encuentre de los archivos terraform
  - "terraform fmt terraform.tf", Hace el arreglo de la indentación de solo el archivo definido

```
jonatan@ubuntu:~/practica_terraform/practica_04$ terraform fmt terraform.tf
terraform.tf
jonatan@ubuntu:~/practica_terraform/practica_04$ cat terraform.tf
resource "local_file" "cali" {
  content = "Hola, grupo devops 2024, este es un ejemplo de terraform"
  filename = "seti.txt"
}
jonatan@ubuntu:~/practica_terraform/practica_04$
```

7. Ahora vamos a ejecutar el comando "terraform validate", lo que hace este comando es validar si la sintaxis del archivo es correcta.
- "terraform validate", Hace una validación a nivel general en la carpeta en la que se encuentre de los archivos terraform.

```
jonatan@ubuntu:~/practica_terraform/practica_04$ terraform validate
Success! The configuration is valid.
jonatan@ubuntu:~/practica_terraform/practica_04$
```

8. Ahora vamos a editar el archivo "terraform.tf", y haremos que se genere un error.
- nano terraform.tf

```
GNU nano 6.2
resource "local_file" "cali" {
  content = "Hola, grupo devops 2024, este es un ejemplo de terraform"
  filename "seti.txt"
}
```

- Quitamos el igual despues de filename.
- Control + O
- Control + x



9. Ahora comprobamos de nuevo con el comando "terraform validate".

```
jonatan@ubuntu:~/practica_terraform/practica_04$ terraform validate

Error: Invalid block definition

on terraform.tf line 3, in resource "local_file" "cali":
  3:   filename "seti.txt"
  4: }

A block definition must have block content delimited by "{" and "}", starting on the same line as the block header.

jonatan@ubuntu:~/practica_terraform/practica_04$
```

10. Como vimos nos dice el archivo, el recurso y la línea del error para solucionarlo.

**Nota:** En proyecto, antes de ejecutar "terraform plan", es recomendado ejecutar "terraform validate" para validar su sintaxis correcta.

11. Arreglamos el error que generamos anteriormente con los comandos antes utilizados, y hacemos el "terraform validate" y comprobamos que todo esté bien.

```
jonatan@ubuntu:~/practica_terraform/practica_04$ cat terraform.tf
resource "local_file" "cali" {
  content = "Hola, grupo devops 2024, este es un ejemplo de terraform"
  filename = "seti.txt"
}
jonatan@ubuntu:~/practica_terraform/practica_04$ terraform validate
Success! The configuration is valid.
jonatan@ubuntu:~/practica_terraform/practica_04$
```