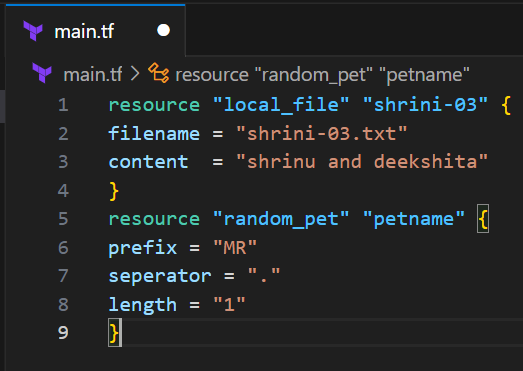
**Terraform-02**

****

**If we have Multiple Providers:**

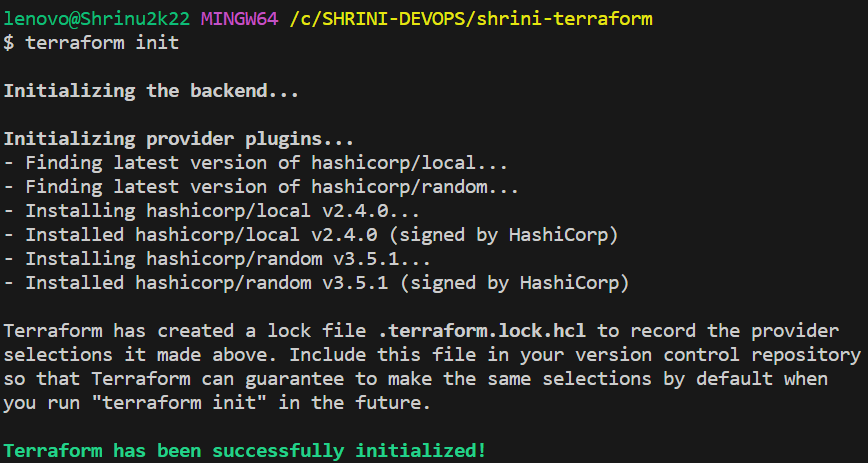
**“We can create a new dir and create a main.tf or else we can add the resources on the existing code”**

****

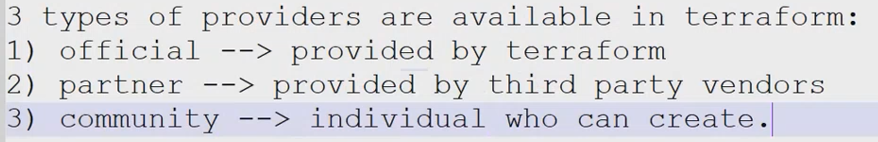
**{ In 1st provider: “local\_file” we are creating a file in our local**

**In 2nd provider: “random\_pet” we are creating a random name**

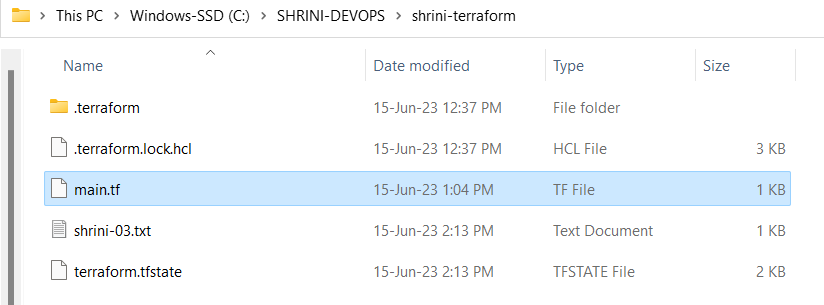
**\*whenever we add any new providers we have to run “terraform init” for download the dependencies (plugins, API,etc) by creating .terraform folder }**

****

* **It installed provider with the version of: “local=2.4.0 and random=3.5.1”**
* **terraform always try to installed latest version**
* **And if we want to change the provider version we can configure  
  so why we need any specific provider version;   
  “if any application having dependences for the compatibility purpose then we have to maintain specific version” like tomcat**

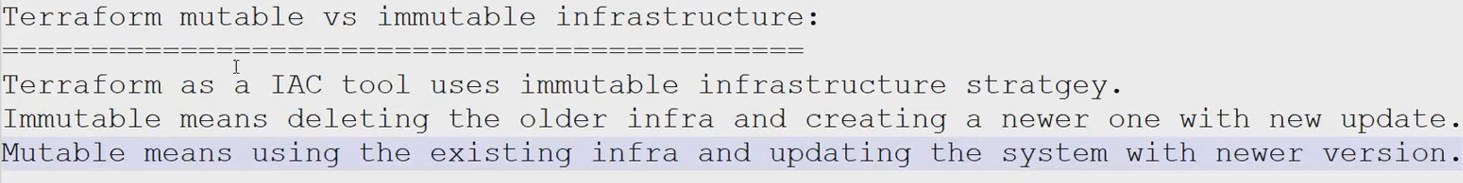
****

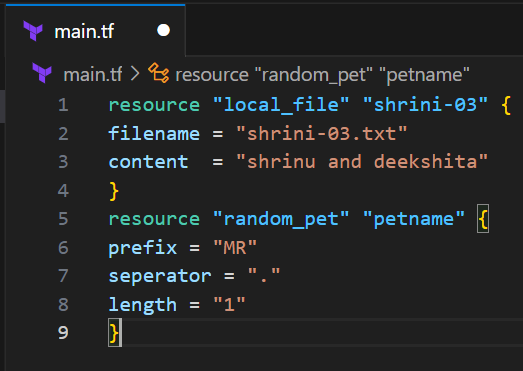
**When we execute “terraform apply” it will create “state.file”**

**it consists the complete metadata of our file like(versions,filename,permissions etc)  
it has the complete information of our infrastructure**

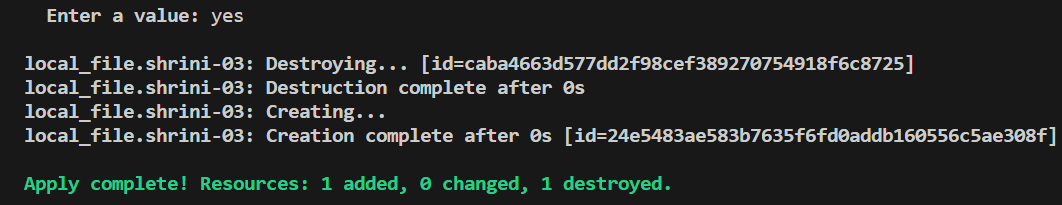
**“In state.tf file we should not make any changes”**

**Terraform infrastructure:**

****

** **

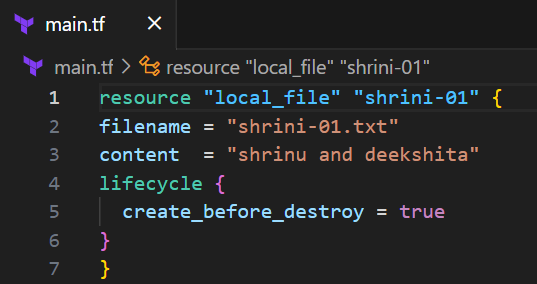
**Here I am change the content in pic-01 so we can see in below pic that it is deleting first and then creating new resources;**

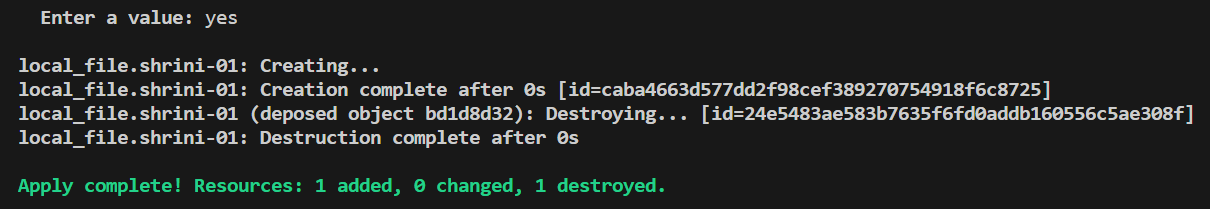
****

**In above we are seeing the file is deleted and creating new resources  
And, if I want to create the file and then delete it,**

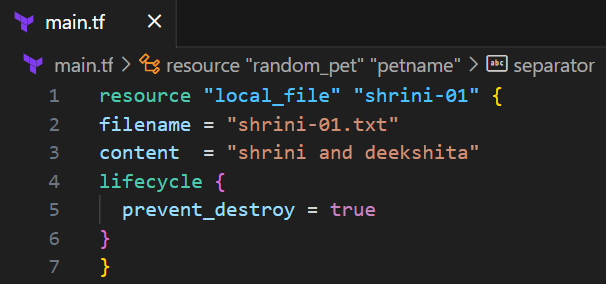
**we have three life-cycle rules;   
lifecycle will be for whole resource**

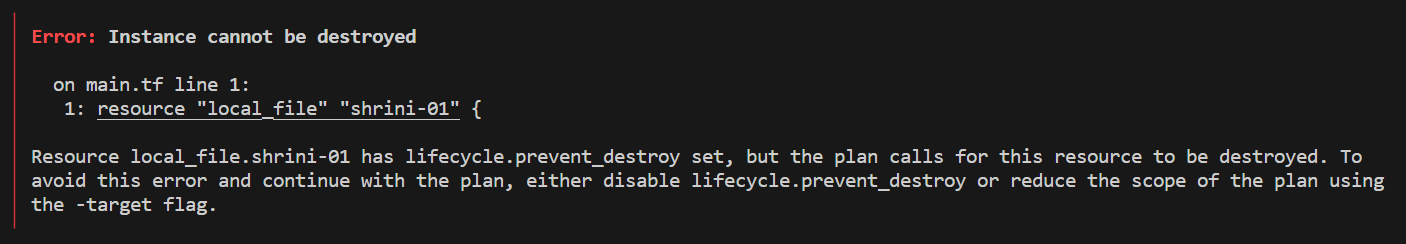
**1. create\_before\_destroy : (works on apply command)**

****

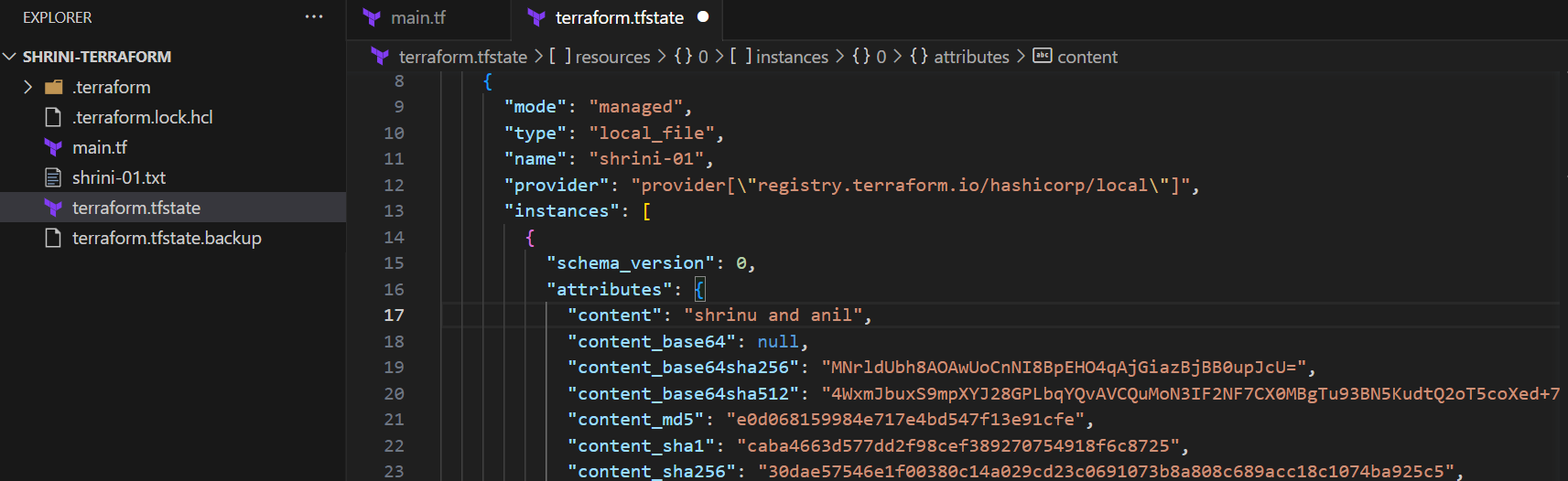
****

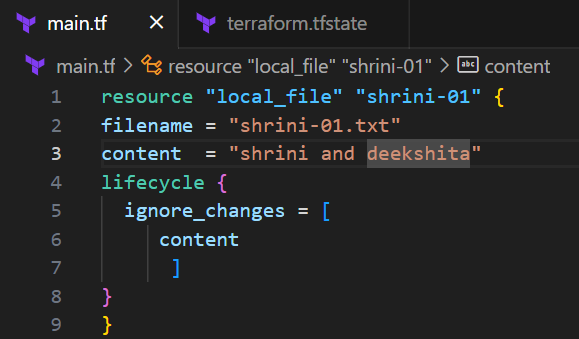
**2. prevent\_destroy : (works on destroy command)**

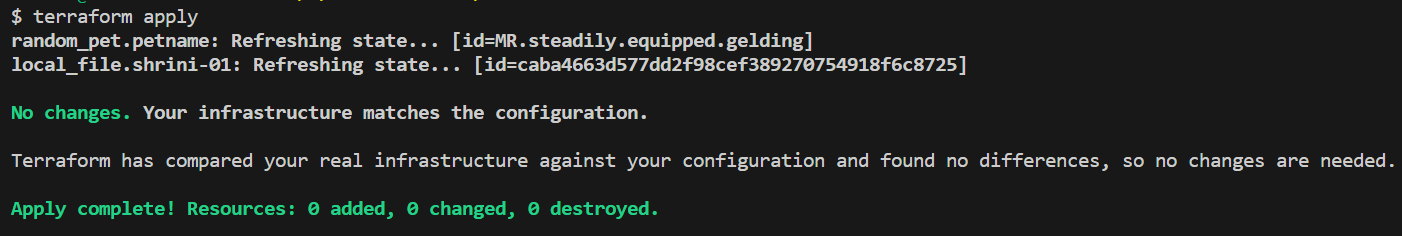
****

****

**3. Ignore\_changes : (works on apply command and works between state.tf and main.tf)**

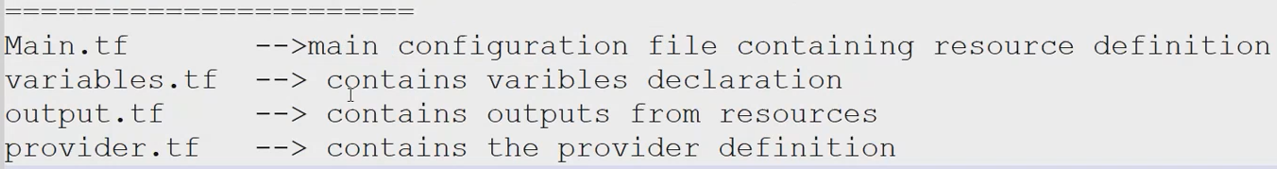
****

****

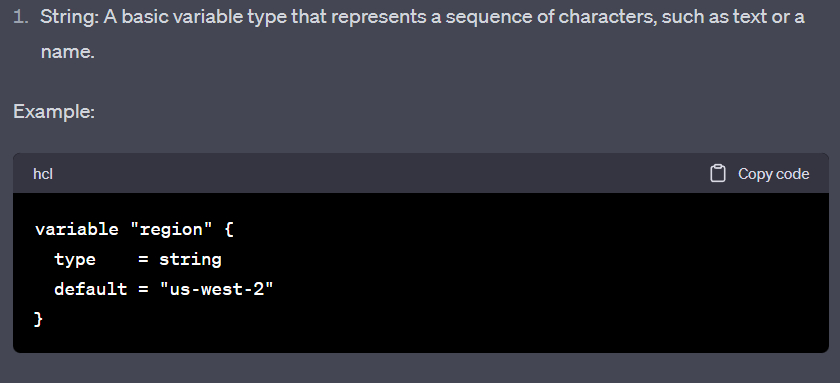
****

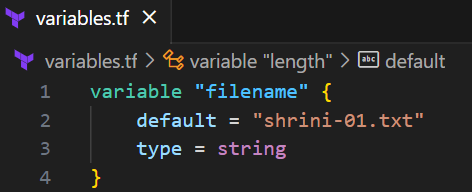
**{While executing apply command it is not changing in our main.tf file with the help of “ignore\_changes”}**

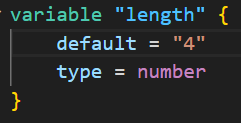
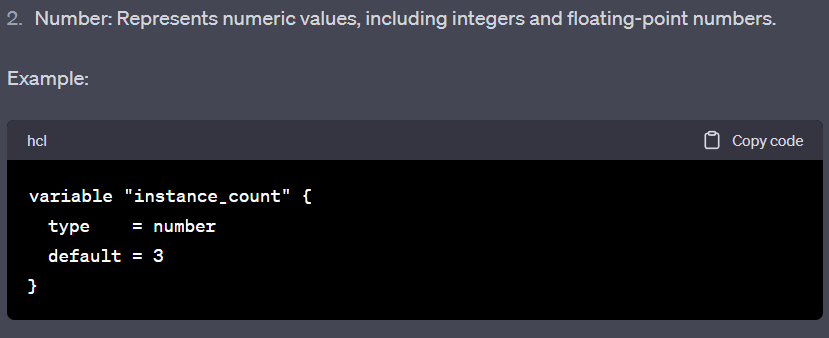
**\*\* CONFIGURATIONS DIRECTORY:**

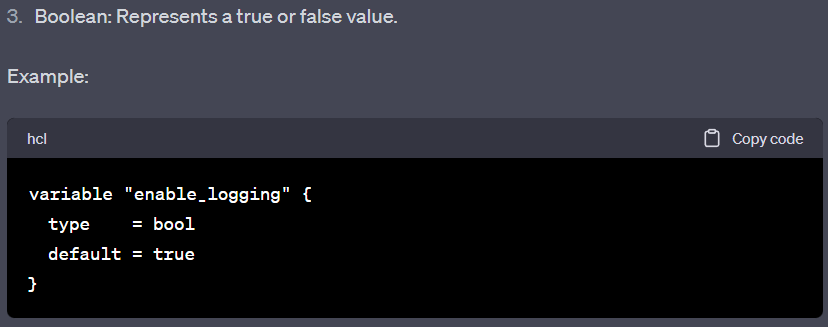
****

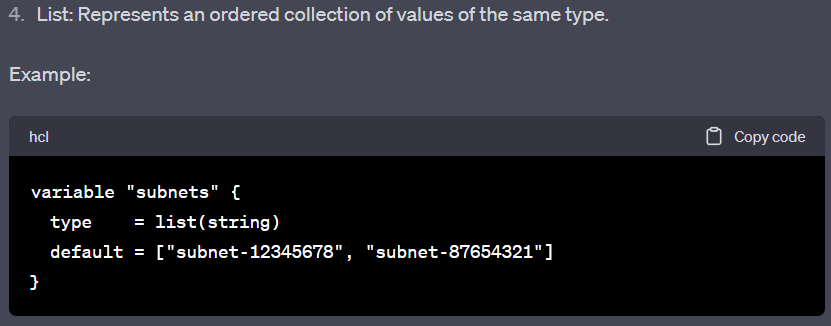
**Variables.tf:**

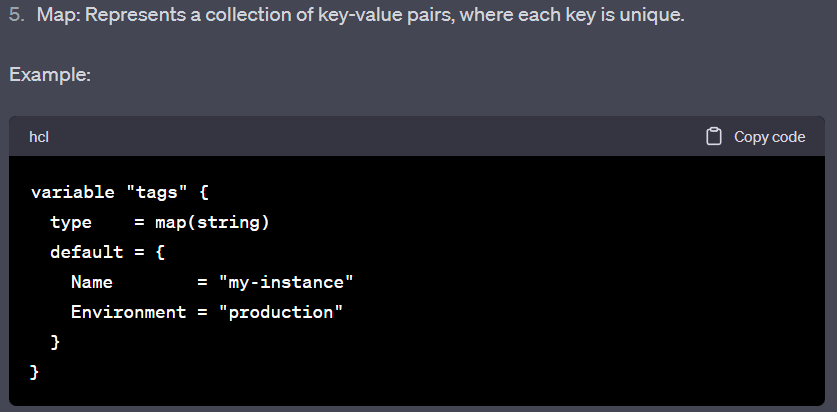
****

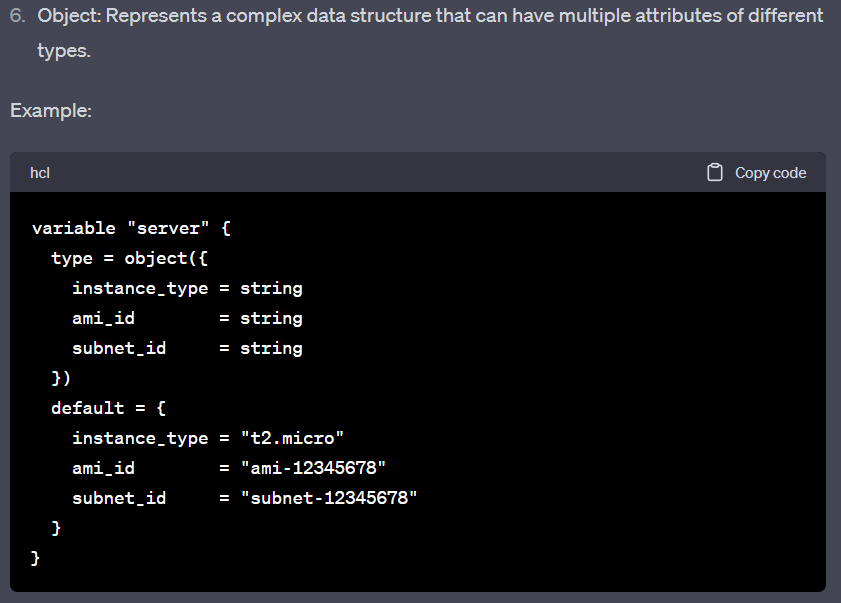
****

****

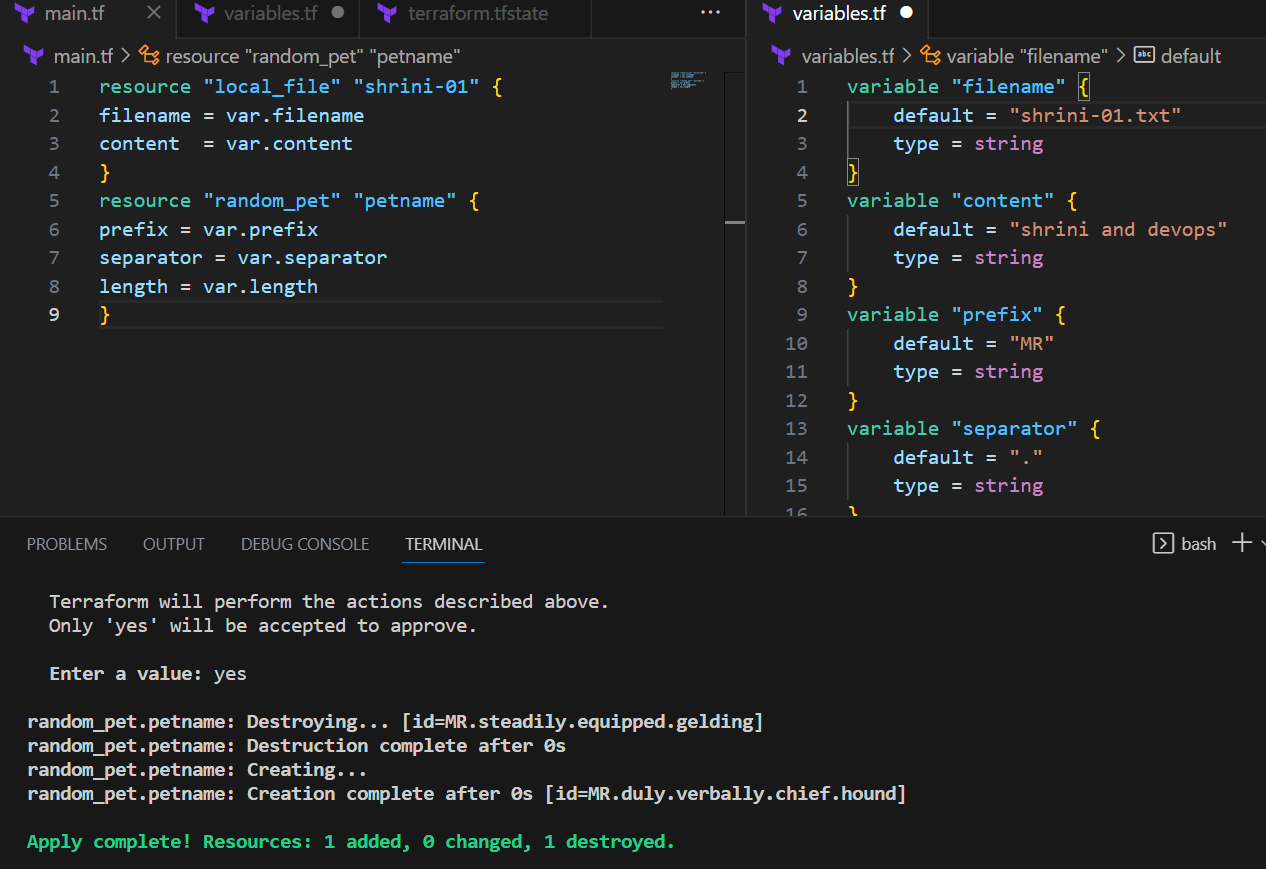
****

****

****

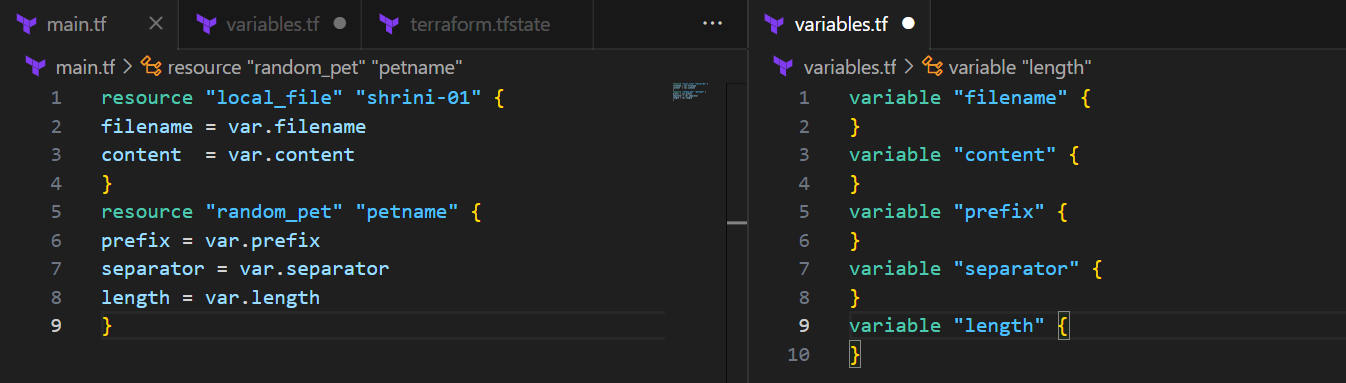
****

**“So we have six types of variables in (tf) as mentioned above”**

****

**\*\*Command line arguments:**

**If any user created any variable and did not declared any variable then terraform consider as “command line Arguments” it will ask the user to input at the run-time**

****

**We can push our variables at the run time by the command;  
-------------------------------------------------------------------------------------------------------------------------------**