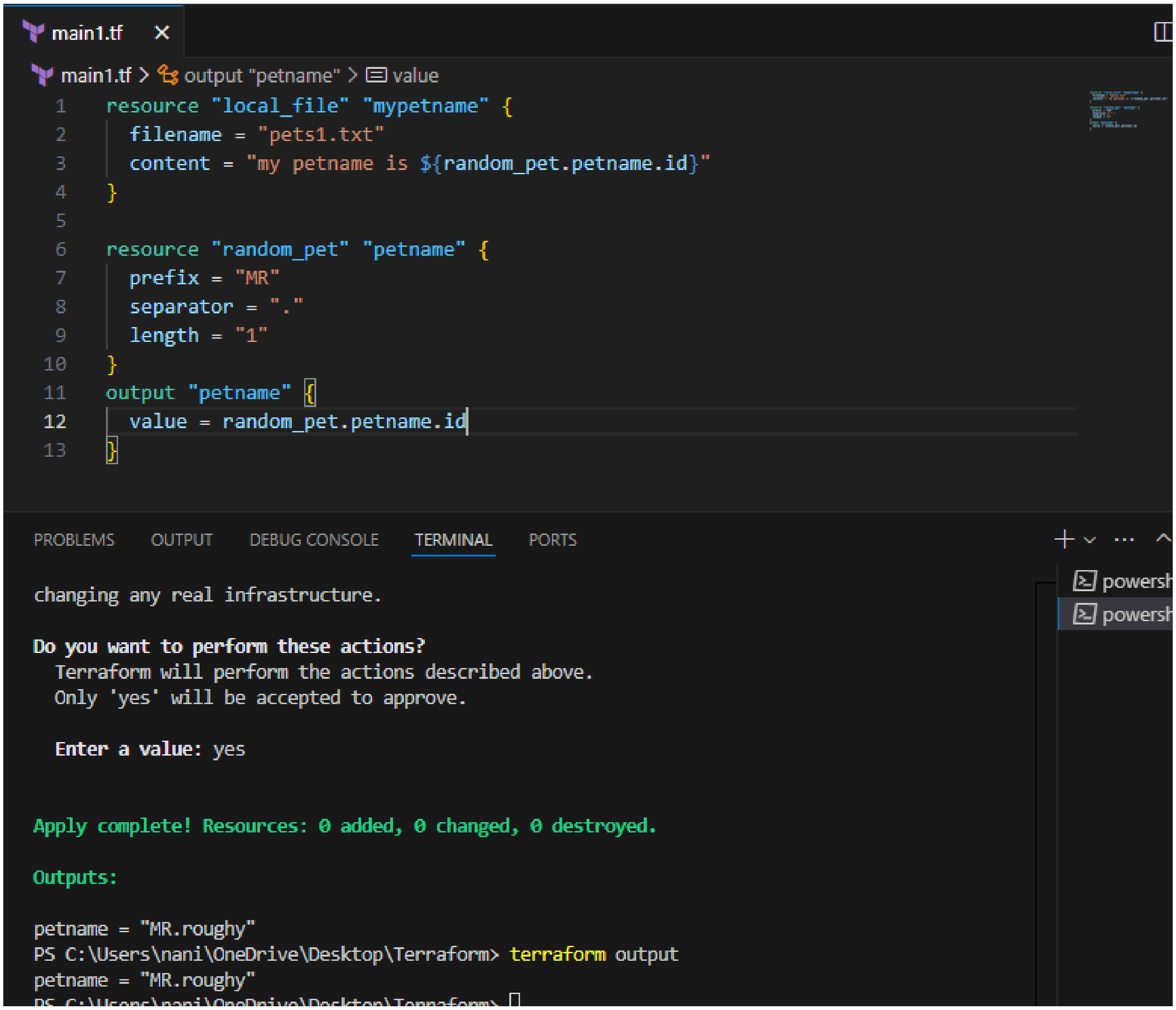
JENKINS-3&4

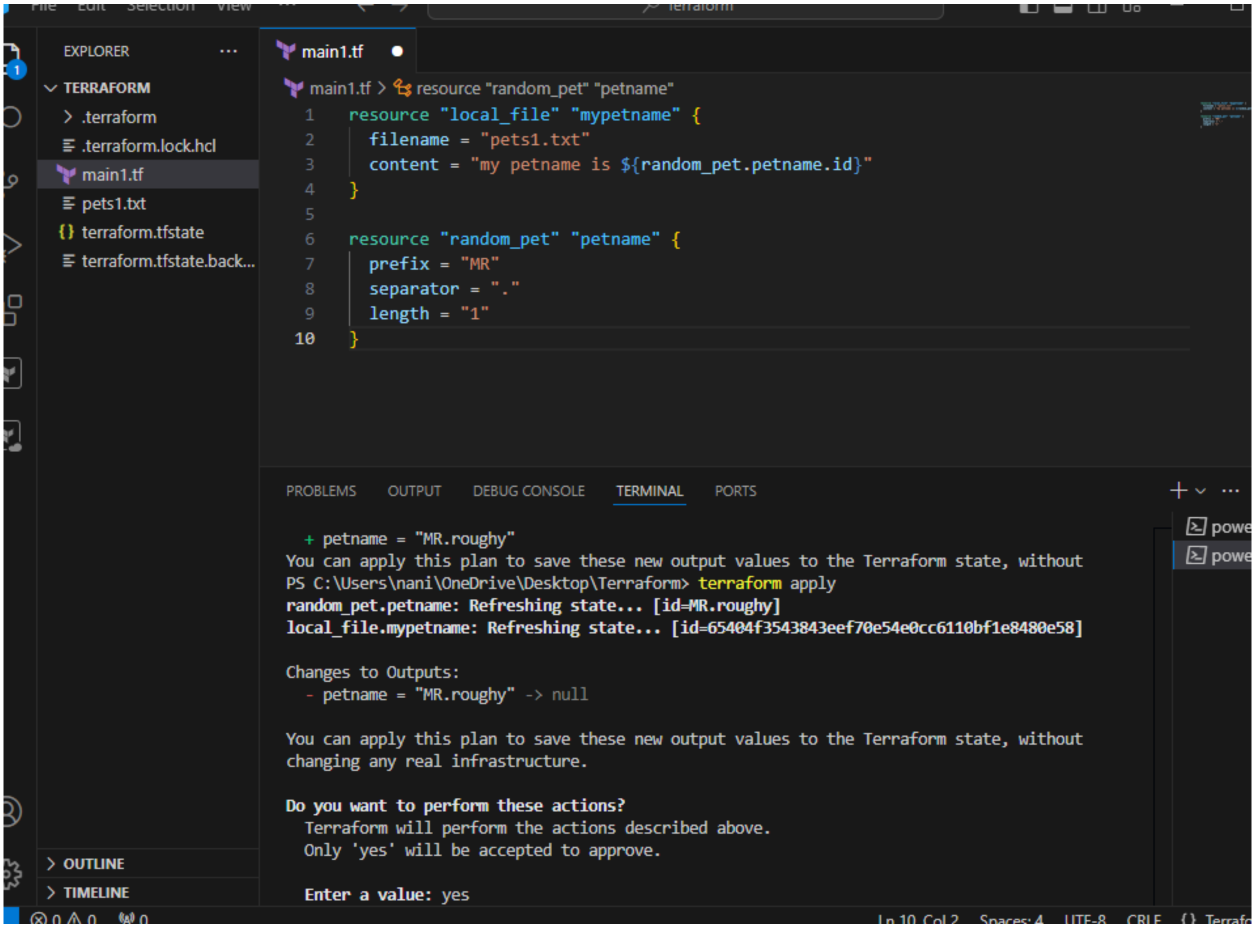
1) Watch terraform-03 video.

2) Execute the script shown in video.



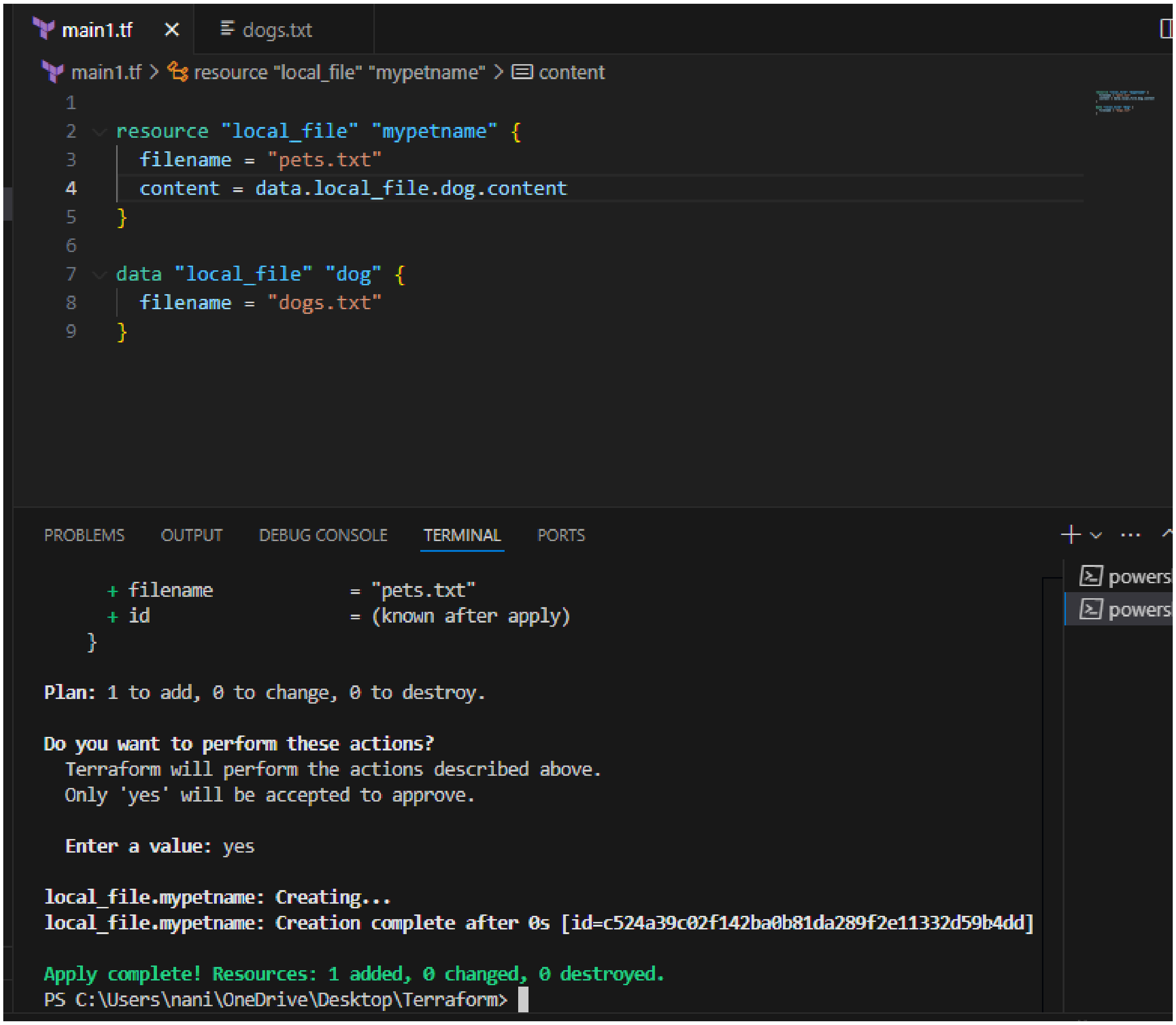
```
resource "local_file" "mypetname" {
  filename = "pets1.txt"
  content = "my petname is ${random_pet.petname.id}"
}

resource "random_pet" "petname" {
  prefix = "MR"
  separator = "."
  length = "1"
}
output "petname" {
  value = random_pet.petname.id
}
```

```
changing any real infrastructure.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes


Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

petname = "MR.roughy"
PS C:\Users\nani\OneDrive\Desktop\Terraform> terraform output
petname = "MR.roughy"
PS C:\Users\nani\OneDrive\Desktop\Terraform>
```



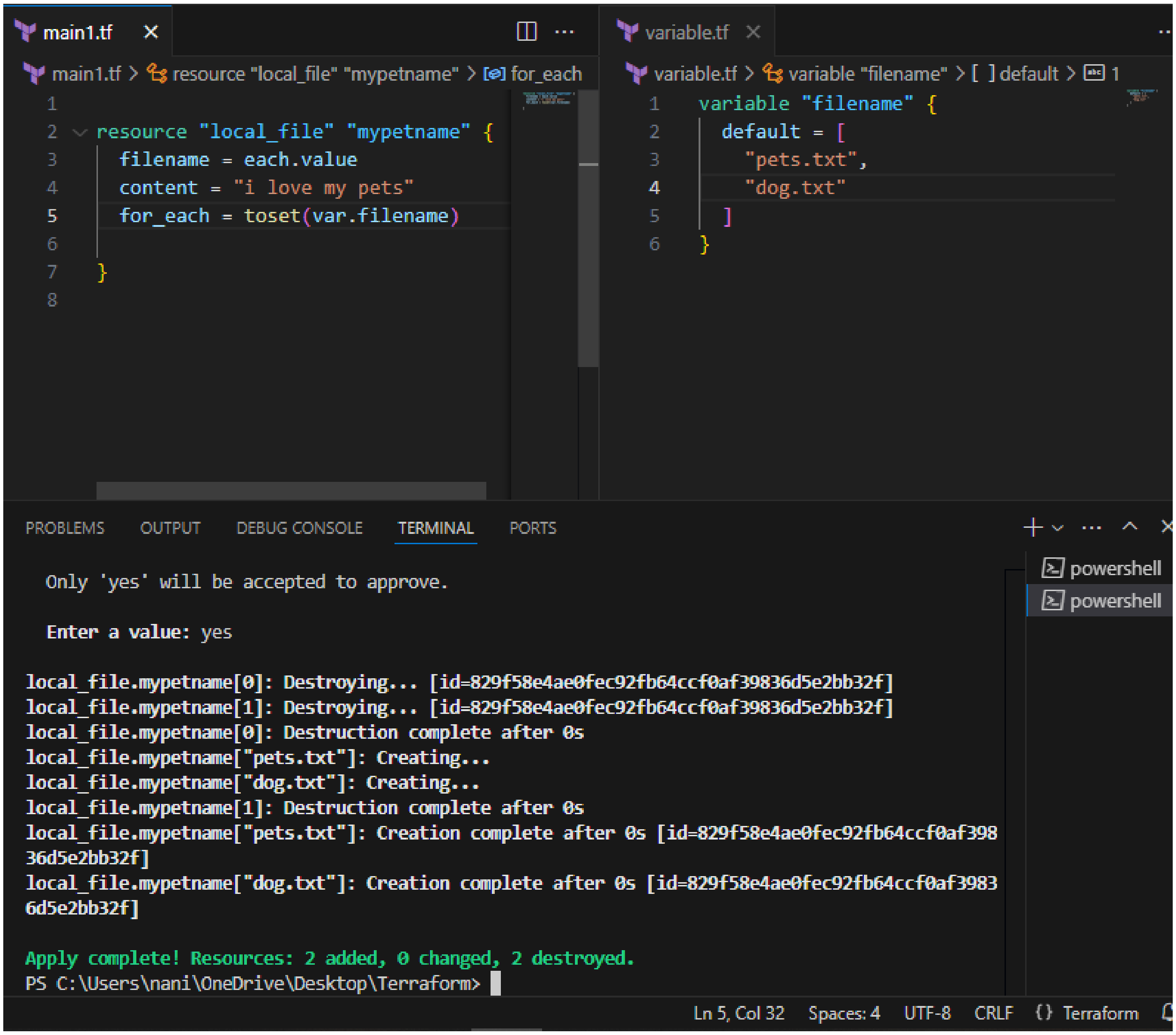
```
resource "local_file" "mypetname" {
  filename = "pets1.txt"
  content = "my petname is ${random_pet.petname.id}"
}

resource "random_pet" "petname" {
  prefix = "MR"
  separator = "."
  length = "1"
}
```

```
  + petname = "MR.roughy"
You can apply this plan to save these new output values to the Terraform state, without
PS C:\Users\nani\OneDrive\Desktop\Terraform> terraform apply
random_pet.petname: Refreshing state... [id=MR.roughy]
local_file.mypetname: Refreshing state... [id=65404f3543843eef70e54e0cc6110bf1e8480e58]

Changes to Outputs:
  - petname = "MR.roughy" -> null

You can apply this plan to save these new output values to the Terraform state, without
changing any real infrastructure.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes
```



```

resource "local_file" "mypetname" {
  filename = "pets.txt"
  content = data.local_file.dog.content
}

data "local_file" "dog" {
  filename = "dogs.txt"
}
```

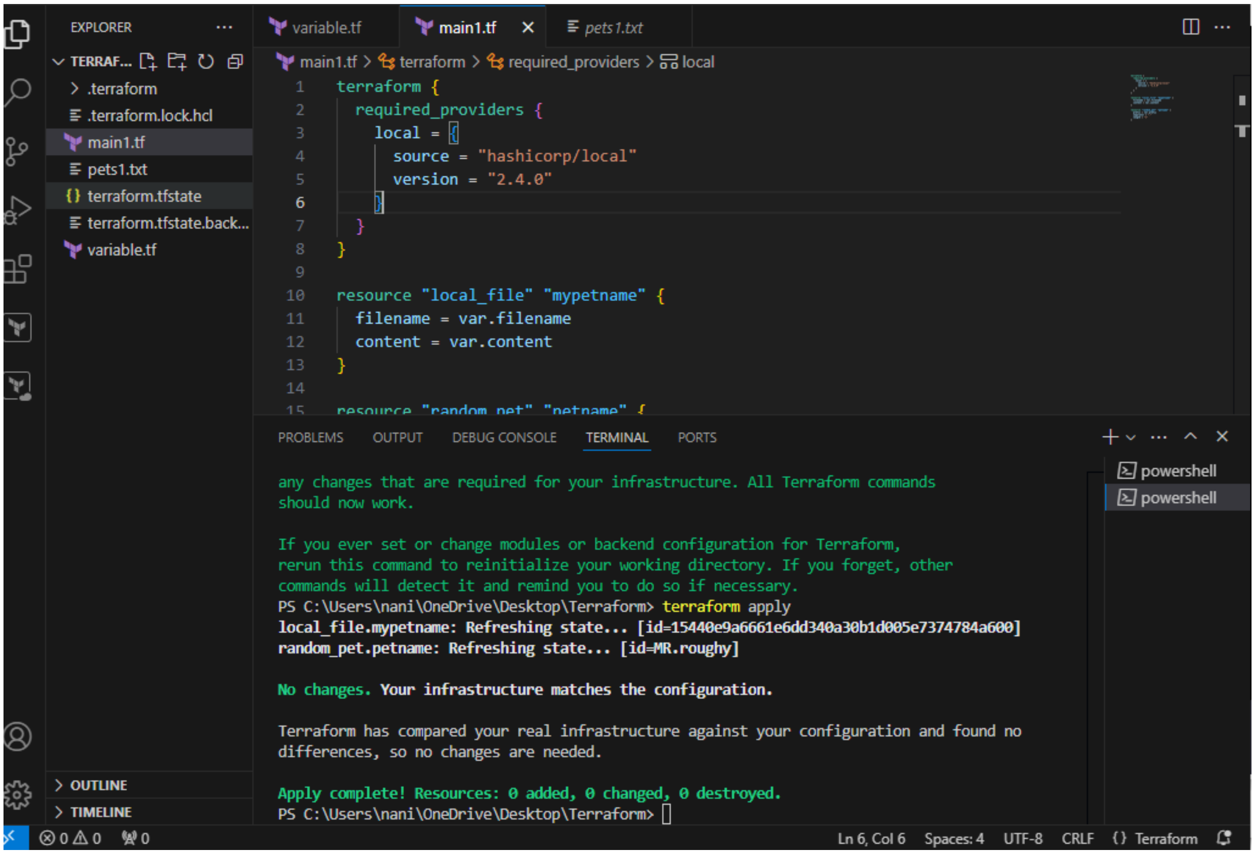
```
      + filename             = "pets.txt"
      + id                   = (known after apply)
    }

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

local_file.mypetname: Creating...
local_file.mypetname: Creation complete after 0s [id=c524a39c02f142ba0b81da289f2e11332d59b4dd]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
PS C:\Users\nani\OneDrive\Desktop\Terraform>
```



```

resource "local_file" "mypetname" {
  filename = each.value
  content = "i love my pets"
  for_each = toset(var.filename)

}

```

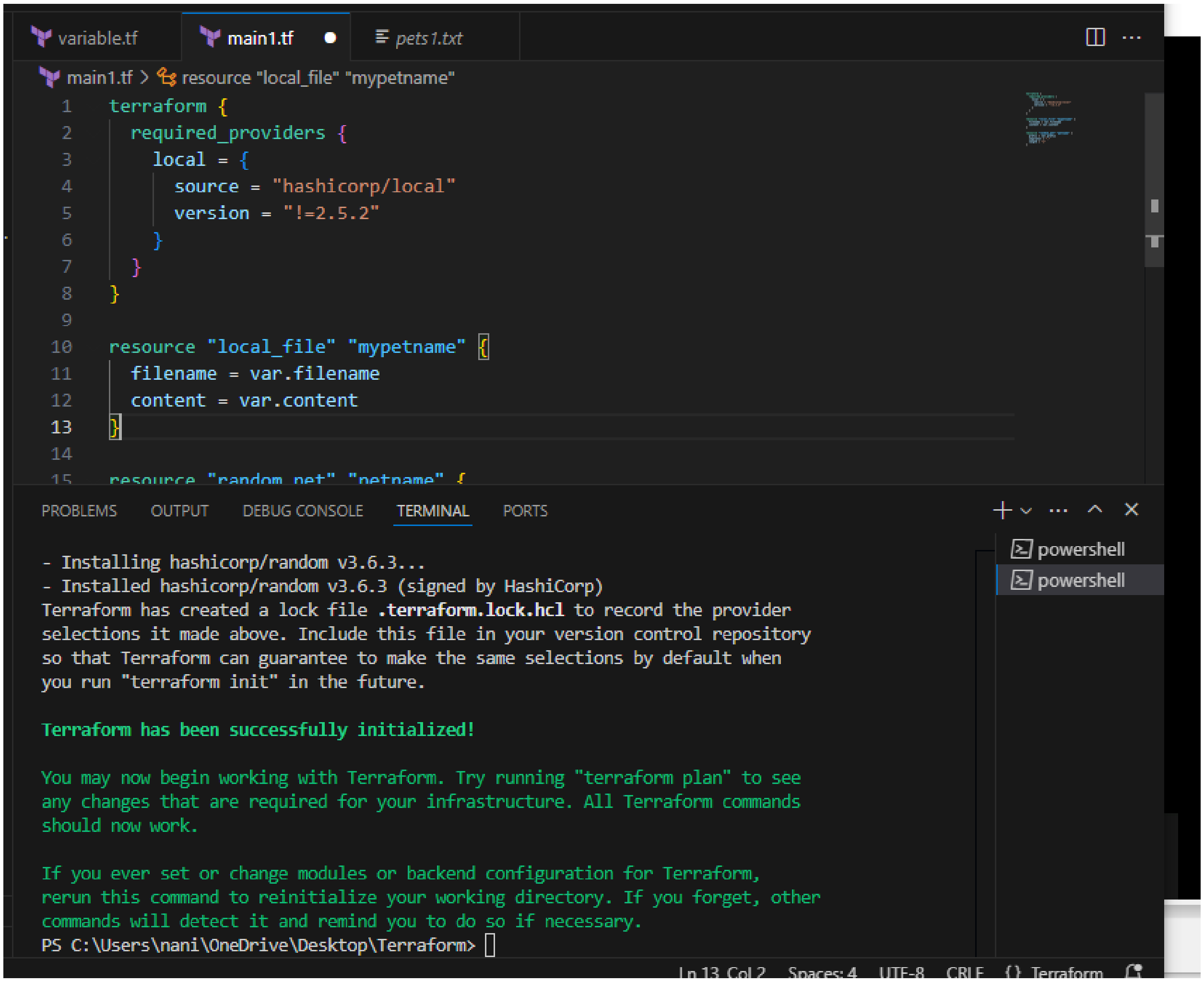
```
variable "filename" {
  default = [
    "pets.txt",
    "dog.txt"
  ]
}
```

```
  Only 'yes' will be accepted to approve.

  Enter a value: yes

local_file.mypetname[0]: Destroying... [id=829f58e4ae0fec92fb64ccf0af39836d5e2bb32f]
local_file.mypetname[1]: Destroying... [id=829f58e4ae0fec92fb64ccf0af39836d5e2bb32f]
local_file.mypetname[0]: Destruction complete after 0s
local_file.mypetname["pets.txt"]: Creating...
local_file.mypetname["dog.txt"]: Creating...
local_file.mypetname[1]: Destruction complete after 0s
local_file.mypetname["pets.txt"]: Creation complete after 0s [id=829f58e4ae0fec92fb64ccf0af398
36d5e2bb32f]
local_file.mypetname["dog.txt"]: Creation complete after 0s [id=829f58e4ae0fec92fb64ccf0af3983
6d5e2bb32f]

Apply complete! Resources: 2 added, 0 changed, 2 destroyed.
PS C:\Users\nani\OneDrive\Desktop\Terraform>
```



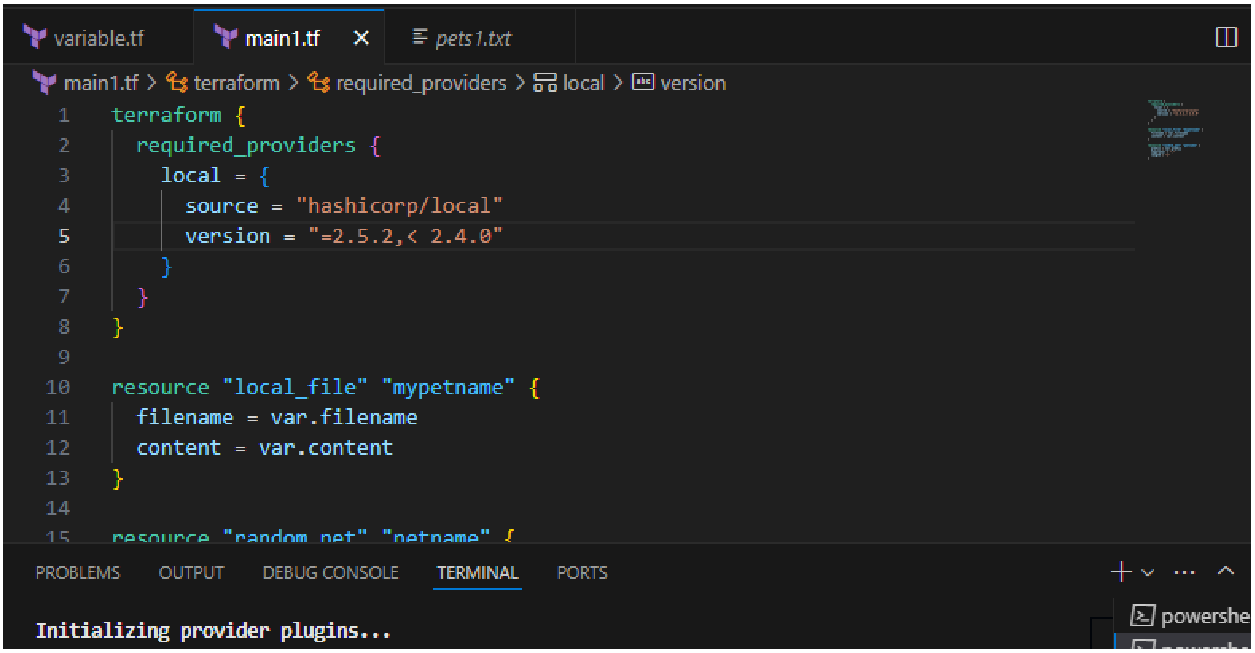
```
terraform {
  required_providers {
    local = {
      source = "hashicorp/local"
      version = "2.4.0"
    }
  }
}

resource "local_file" "mypetname" {
  filename = var.filename
  content = var.content
}

resource "random_pet" "petname" {
```

```
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
PS C:\Users\nani\OneDrive\Desktop\Terraform> terraform apply
local_file.mypetname: Refreshing state... [id=15440e9a6661e6dd340a30b1d005e7374784a600]
random_pet.petname: Refreshing state... [id=MR.roughy]

No changes. Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no
differences, so no changes are needed.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.
PS C:\Users\nani\OneDrive\Desktop\Terraform> 
```



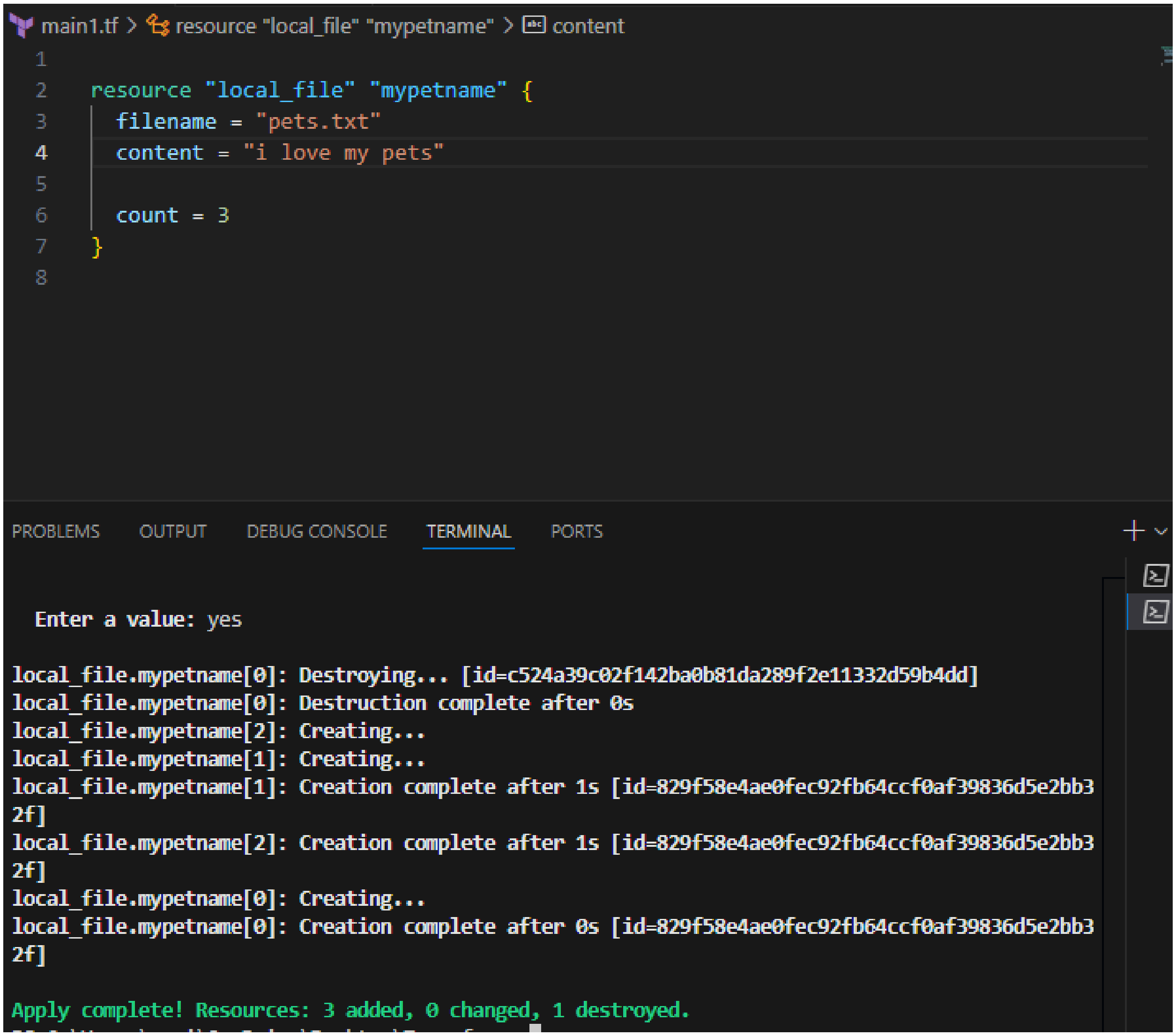
```
terraform {
  required_providers {
    local = {
      source = "hashicorp/local"
      version = "!=2.5.2"
    }
  }
}

resource "local_file" "mypetname" {
  filename = var.filename
  content = var.content
}

resource "random_pet" "petname" {
```

```
- Installing hashicorp/random v3.6.3...
- Installed hashicorp/random v3.6.3 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
PS C:\Users\nani\OneDrive\Desktop\Terraform>
```



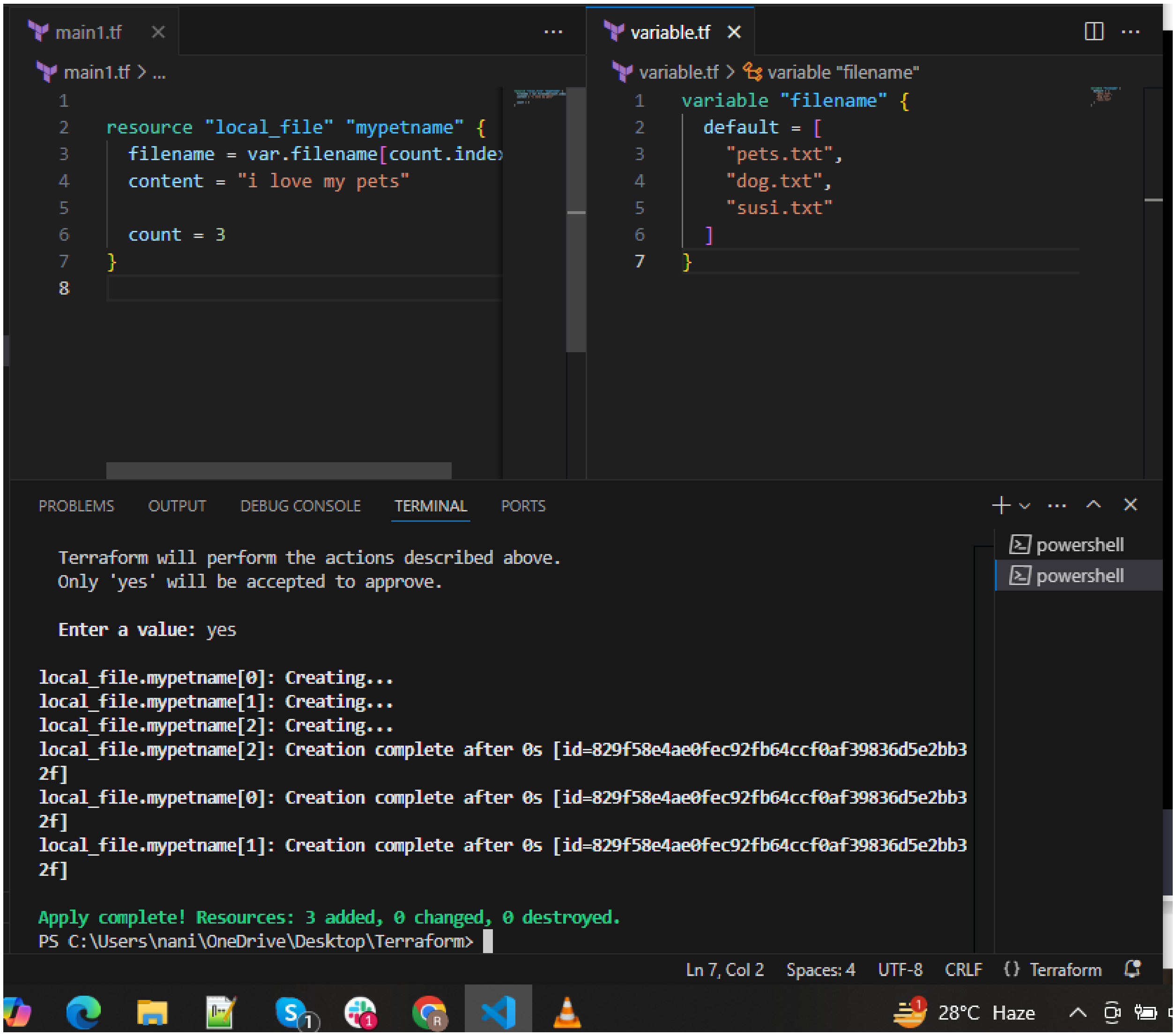
```
terraform {
  required_providers {
    local = {
      source = "hashicorp/local"
      version = "=2.5.2,< 2.4.0"
    }
  }
}

resource "local_file" "mypetname" {
  filename = var.filename
  content = var.content
}

resource "random_pet" "petname" {
```

```
Initializing provider plugins...
```



```

resource "local_file" "mypetname" {
  filename = "pets.txt"
  content = "i love my pets"

  count = 3
}

```

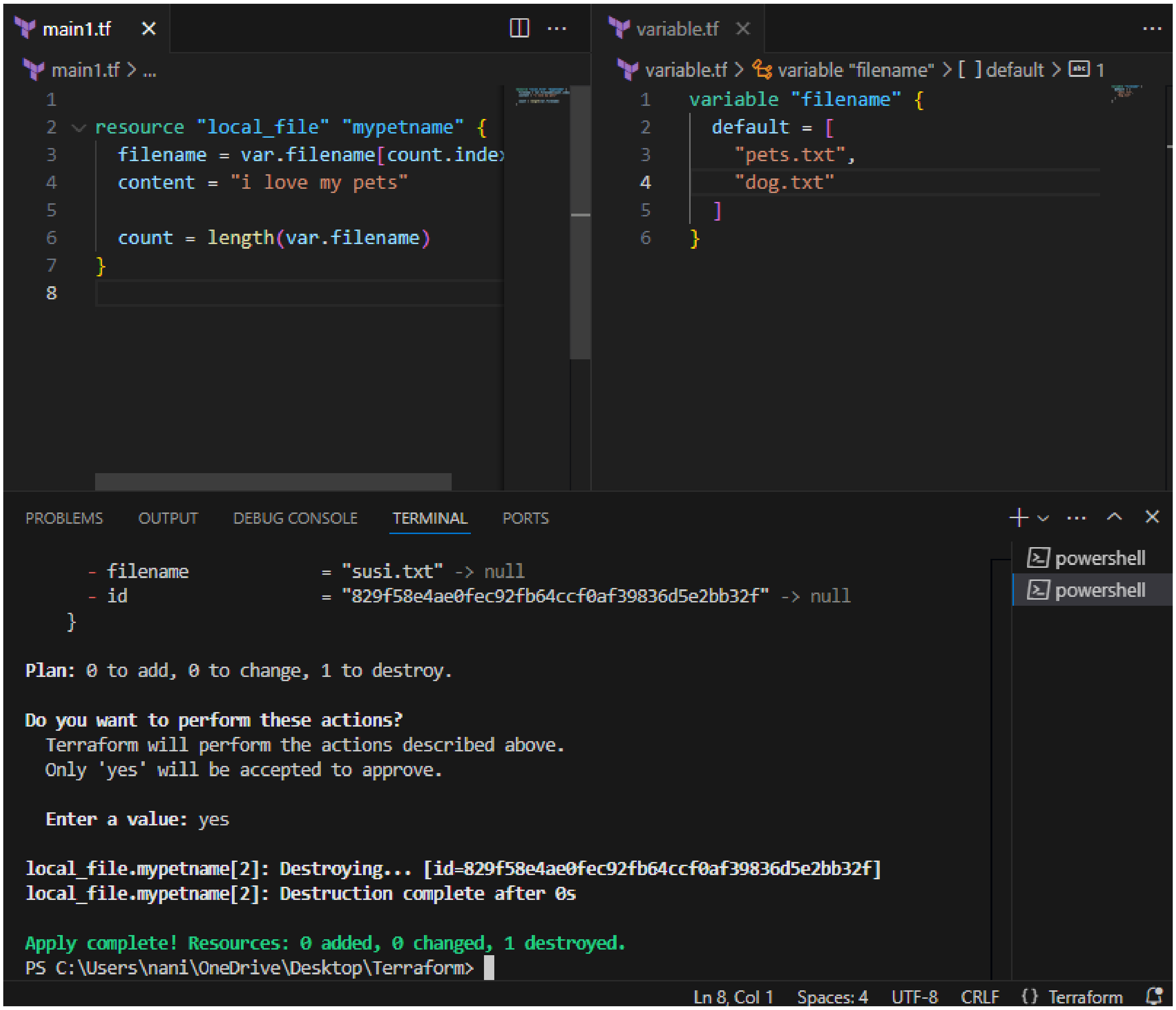
```
  Enter a value: yes

local_file.mypetname[0]: Destroying... [id=c524a39c02f142ba0b81da289f2e11332d59b4dd]
local_file.mypetname[0]: Destruction complete after 0s
local_file.mypetname[2]: Creating...
local_file.mypetname[1]: Creating...
local_file.mypetname[1]: Creation complete after 1s [id=829f58e4ae0fec92fb64ccf0af39836d5e2bb3
2f]
local_file.mypetname[2]: Creation complete after 1s [id=829f58e4ae0fec92fb64ccf0af39836d5e2bb3
2f]
local_file.mypetname[0]: Creating...
local_file.mypetname[0]: Creation complete after 0s [id=829f58e4ae0fec92fb64ccf0af39836d5e2bb3
2f]

Apply complete! Resources: 3 added, 0 changed, 1 destroyed.
```



```

resource "local_file" "mypetname" {
  filename = var.filename[count.index
  content = "i love my pets"

  count = 3
}

```

```
variable "filename" {
  default = [
    "pets.txt",
    "dog.txt",
    "susi.txt"
  ]
}
```

```
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

local_file.mypetname[0]: Creating...
local_file.mypetname[1]: Creating...
local_file.mypetname[2]: Creating...
local_file.mypetname[2]: Creation complete after 0s [id=829f58e4ae0fec92fb64ccf0af39836d5e2bb3
2f]
local_file.mypetname[0]: Creation complete after 0s [id=829f58e4ae0fec92fb64ccf0af39836d5e2bb3
2f]
local_file.mypetname[1]: Creation complete after 0s [id=829f58e4ae0fec92fb64ccf0af39836d5e2bb3
2f]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
PS C:\Users\nani\OneDrive\Desktop\Terraform>
```



main1.tf:

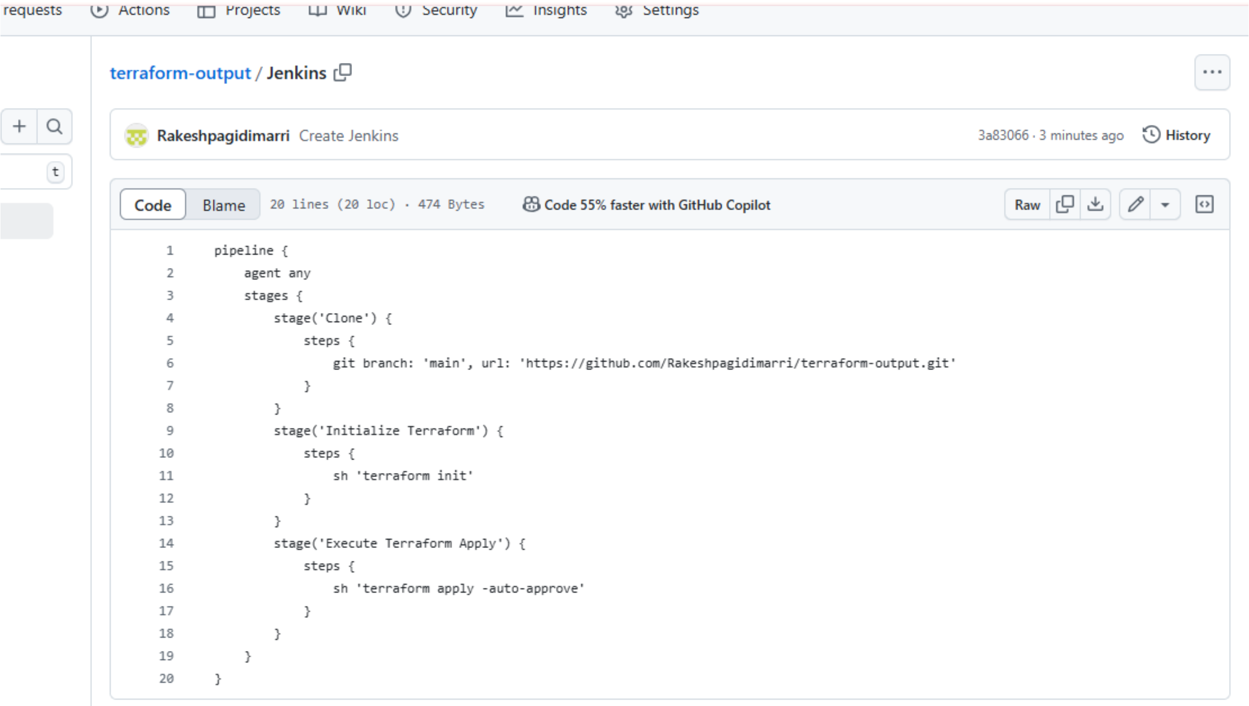
```

resource "local_file" "mypetname" {
  filename = var.filename[count.inde
  content = "i love my pets"

  count = length(var.filename)
}

```

variable.tf:

```
variable "filename" {
  default = [
    "pets.txt",
    "dog.txt"
  ]
}
```

```
      - filename             = "susi.txt" -> null
      - id                   = "829f58e4ae0fec92fb64ccf0af39836d5e2bb32f" -> null
    }

Plan: 0 to add, 0 to change, 1 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

local_file.mypetname[2]: Destroying... [id=829f58e4ae0fec92fb64ccf0af39836d5e2bb32f]
local_file.mypetname[2]: Destruction complete after 0s

Apply complete! Resources: 0 added, 0 changed, 1 destroyed.
PS C:\Users\nani\OneDrive\Desktop\Terraform>
```

3) Intergrate terrafrom in jenkins using Terraform plugin.

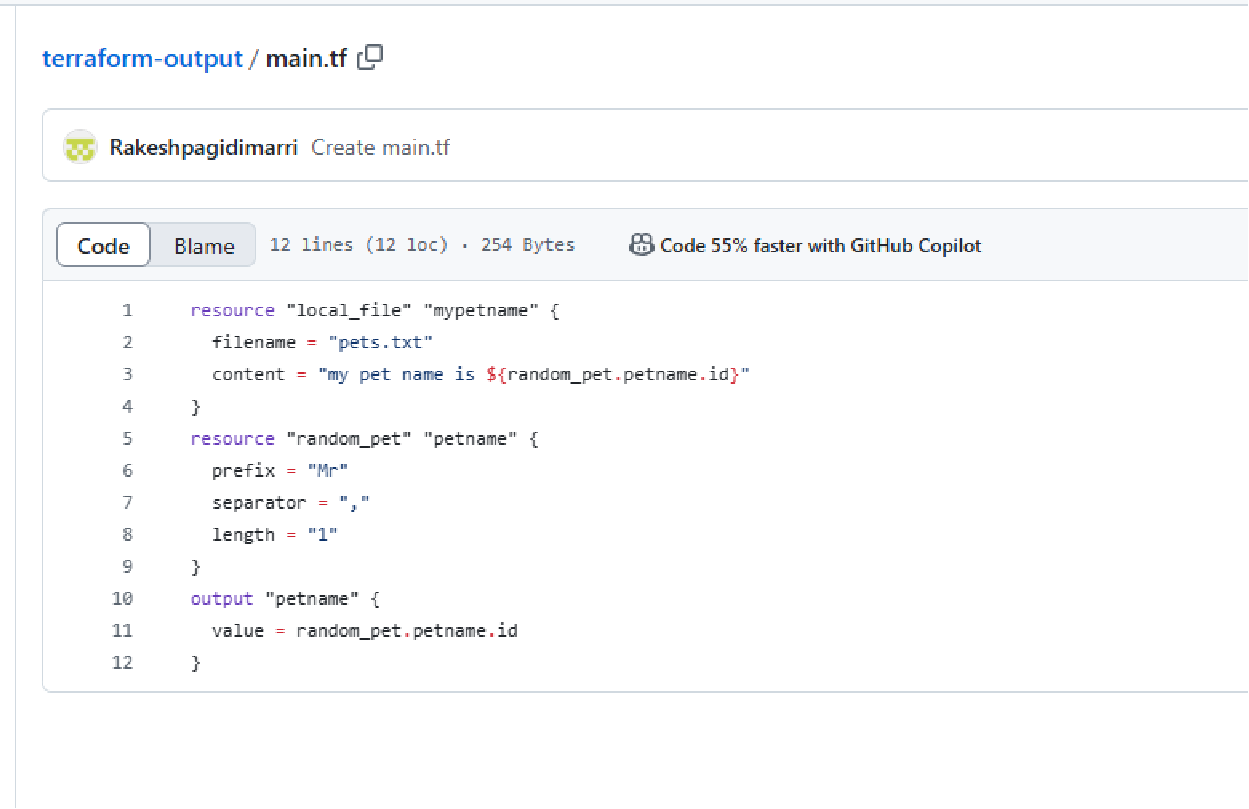


terraform-output / Jenkins

Rakeshpagidimarri Create Jenkins 3a83066 · 3 minutes ago

20 lines (20 loc) · 474 Bytes Code 55% faster with GitHub Copilot

```
pipeline {
    agent any
    stages {
        stage('Clone') {
            steps {
                git branch: 'main', url: 'https://github.com/Rakeshpagidimarri/terraform-output.git'
            }
        }
        stage('Initialize Terraform') {
            steps {
                sh 'terraform init'
            }
        }
        stage('Execute Terraform Apply') {
            steps {
                sh 'terraform apply -auto-approve'
            }
        }
    }
}
```

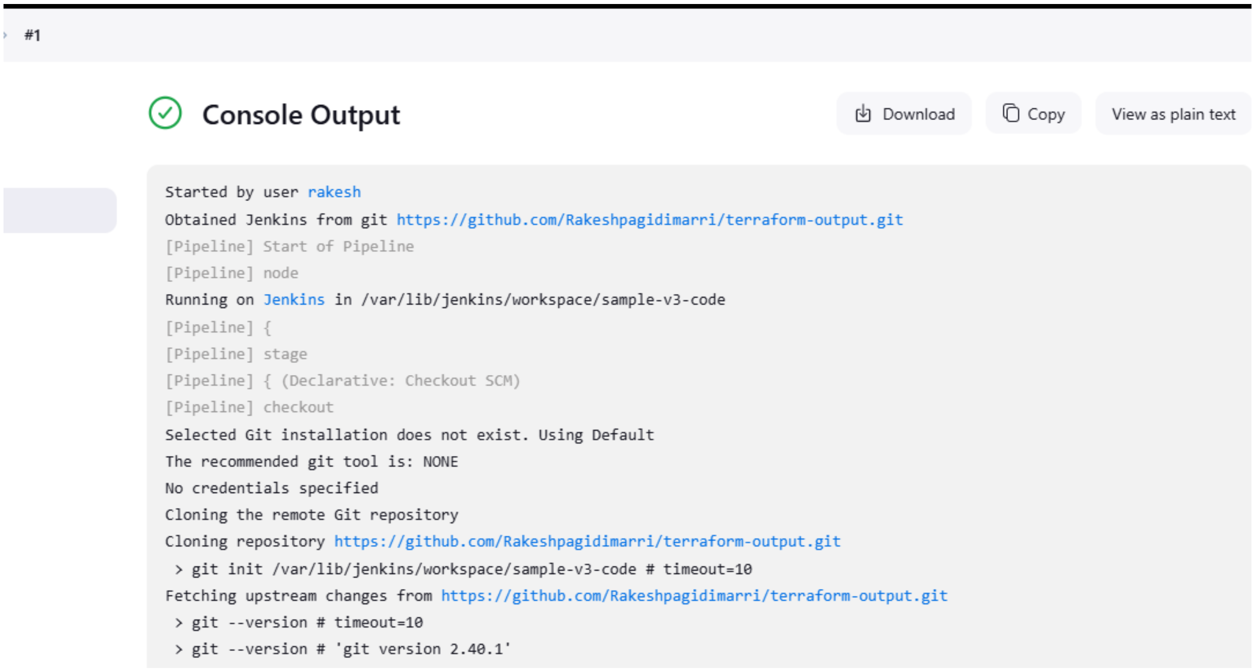


terraform-output / main.tf

Rakeshpagidimarri Create main.tf

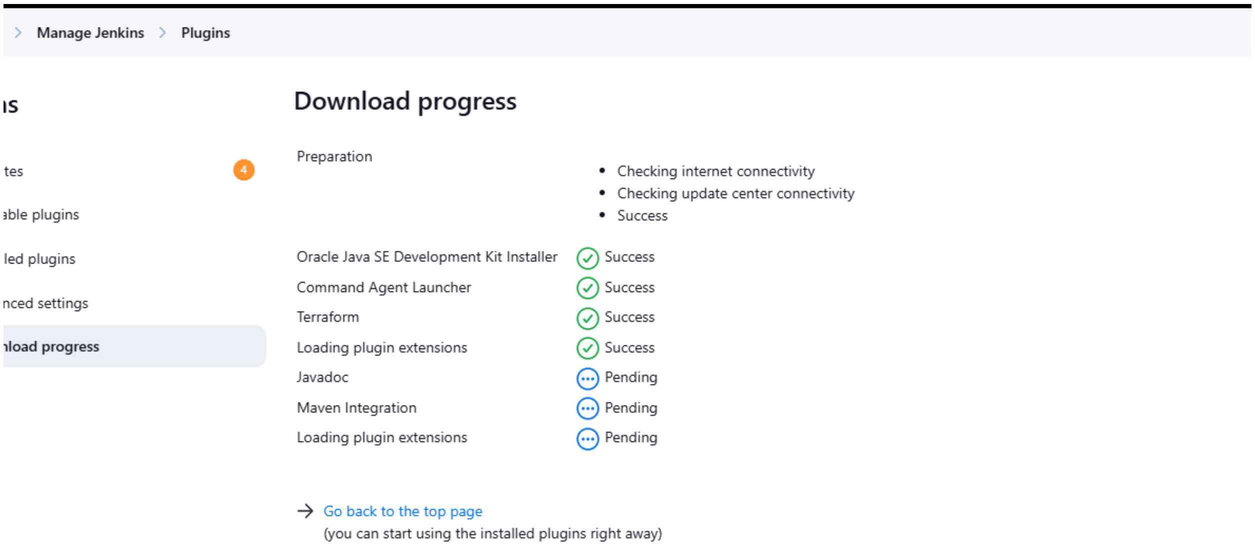
12 lines (12 loc) · 254 Bytes Code 55% faster with GitHub Copilot

```
resource "local_file" "mypetname" {
  filename = "pets.txt"
  content = "my pet name is ${random_pet.petname.id}"
}
resource "random_pet" "petname" {
  prefix = "Mr"
  separator = ","
  length = "1"
}
output "petname" {
  value = random_pet.petname.id
}
```



```
Started by user rakesh
Obtained Jenkins from git https://github.com/Rakeshpagidimarri/terraform-output.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/sample-v3-code
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/Rakeshpagidimarri/terraform-output.git
 > git init /var/lib/jenkins/workspace/sample-v3-code # timeout=10
Fetching upstream changes from https://github.com/Rakeshpagidimarri/terraform-output.git
 > git --version # timeout=10
 > git --version # 'git version 2.40.1'
```

4) Create one jenkins job using MAVEN PROJECT for the below code with two stages. stage 1: Git clone stage 2: Maven Compilation Code: https://github.com/betawins/java-Working-app.git



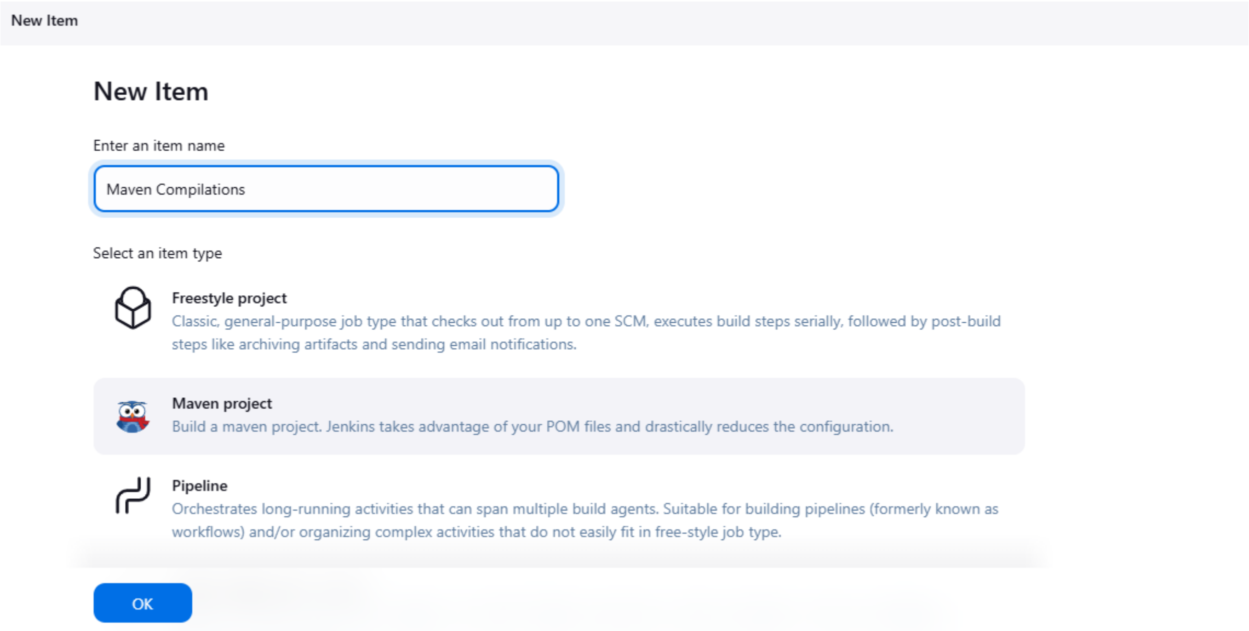
Download progress

Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

Oracle Java SE Development Kit Installer	✓ Success
Command Agent Launcher	✓ Success
Terraform	✓ Success
Loading plugin extensions	✓ Success
Javadoc	⋯ Pending
Maven Integration	⋯ Pending
Loading plugin extensions	⋯ Pending

→ Go back to the top page
(you can start using the installed plugins right away)



New Item

Enter an item name

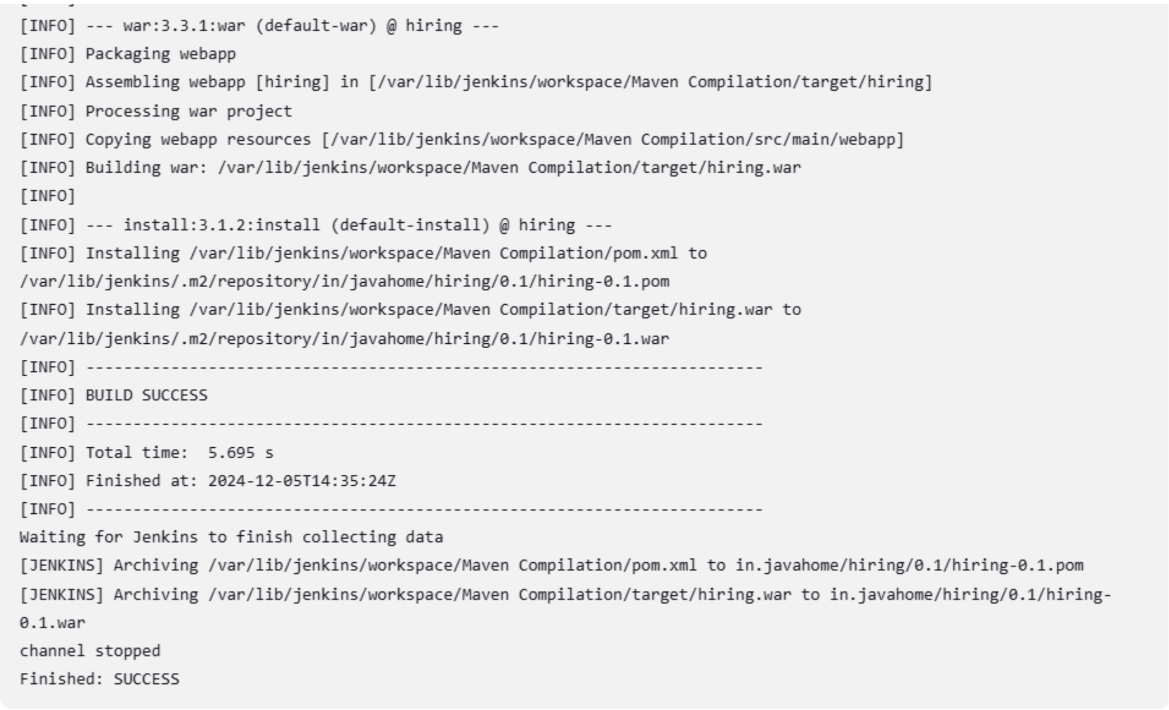
Maven Compilations

Select an item type

Freestyle project
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

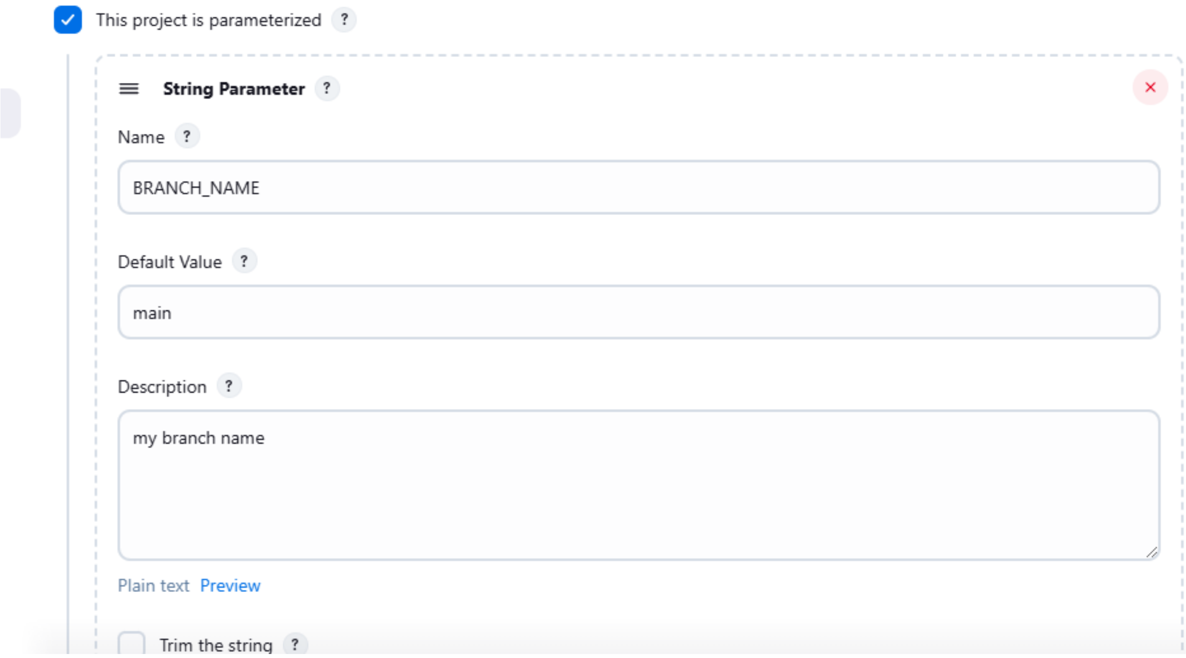
Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.



```
[INFO] --- war:3.3.1:war (default-war) @ hiring ---
[INFO] Packaging webapp
[INFO] Assembling webapp [hiring] in [/var/lib/jenkins/workspace/Maven Compilation/target/hiring]
[INFO] Processing war project
[INFO] Copying webapp resources [/var/lib/jenkins/workspace/Maven Compilation/src/main/webapp]
[INFO] Building war: /var/lib/jenkins/workspace/Maven Compilation/target/hiring.war
[INFO]
[INFO] --- install:3.1.2:install (default-install) @ hiring ---
[INFO] Installing /var/lib/jenkins/workspace/Maven Compilation/pom.xml to
/var/lib/jenkins/.m2/repository/in/javahome/hiring/0.1/hiring-0.1.pom
[INFO] Installing /var/lib/jenkins/workspace/Maven Compilation/target/hiring.war to
/var/lib/jenkins/.m2/repository/in/javahome/hiring/0.1/hiring-0.1.war
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  5.695 s
[INFO] Finished at: 2024-12-05T14:35:24Z
[INFO] ------------------------------------------------------------------------
Waiting for Jenkins to finish collecting data
[JENKINS] Archiving /var/lib/jenkins/workspace/Maven Compilation/pom.xml to in.javahome/hiring/0.1/hiring-0.1.pom
[JENKINS] Archiving /var/lib/jenkins/workspace/Maven Compilation/target/hiring.war to in.javahome/hiring/0.1/hiring-
0.1.war
channel stopped
Finished: SUCCESS
```

5) Use the below code and create a parameterized job in jenkins stage 1: Git clone stage 2: Maven Compilation Code: https://github.com/betawins/java-Working-app.git



☑ This project is parameterized ?

String Parameter ?

Name ?

BRANCH_NAME

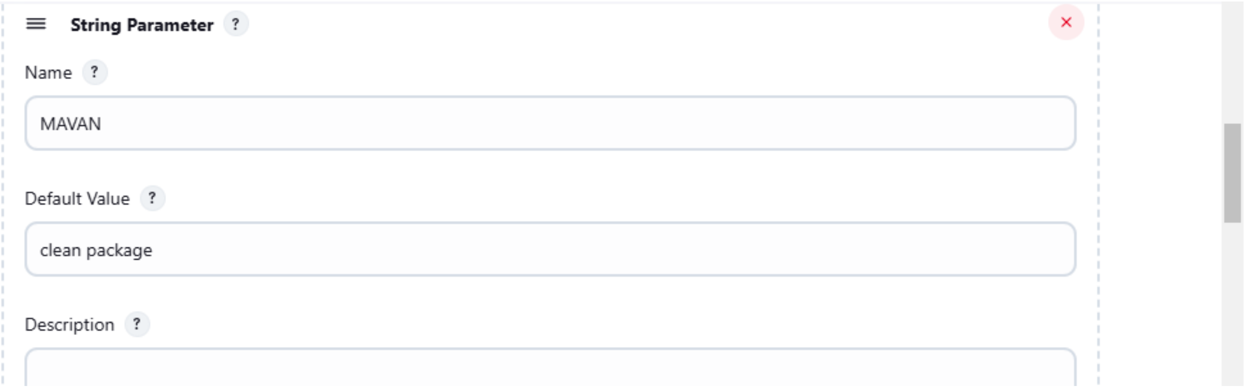
Default Value ?

main

Description ?

my branch name

□ Trim the string ?



String Parameter ?

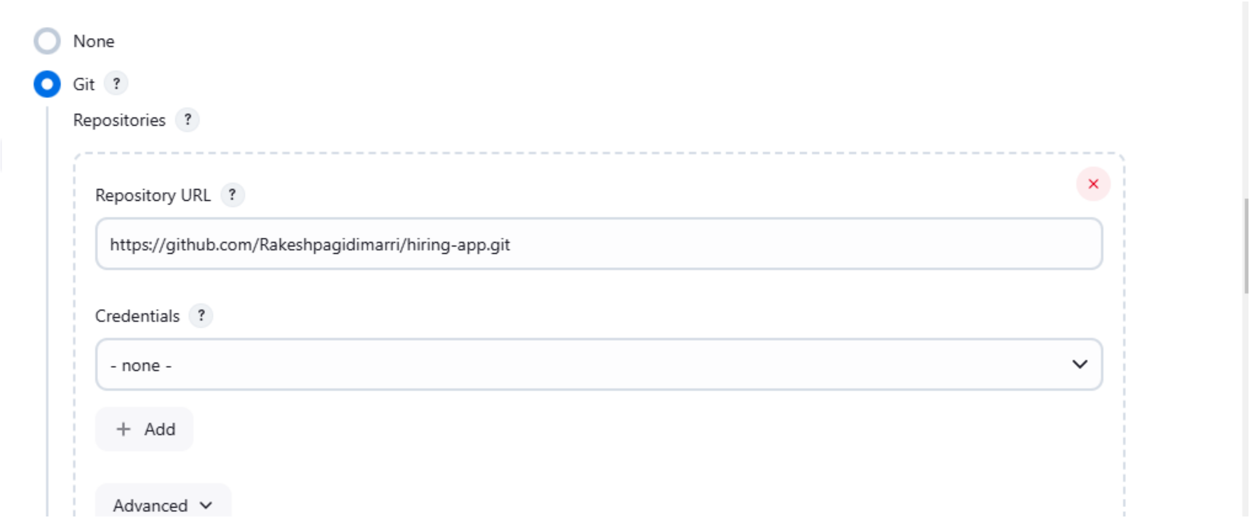
Name ?

MAVAN

Default Value ?

clean package

Description ?



○ None

● Git ?

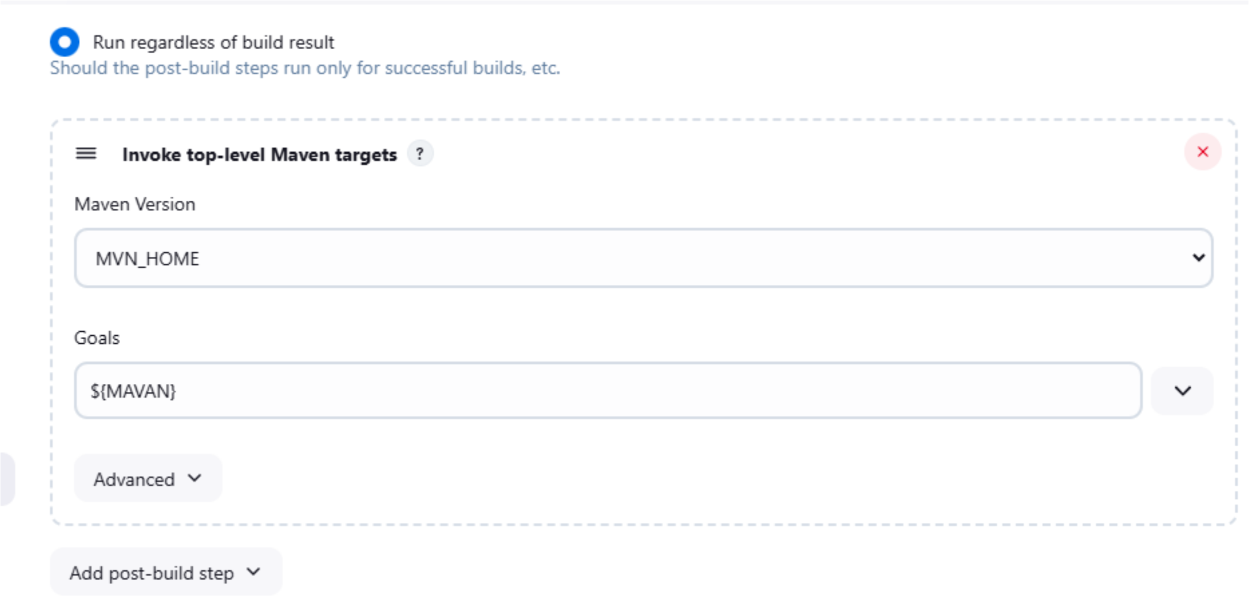
Repositories ?

Repository URL ?

https://github.com/Rakeshpagidimarri/hiring-app.git

Credentials ?

- none -



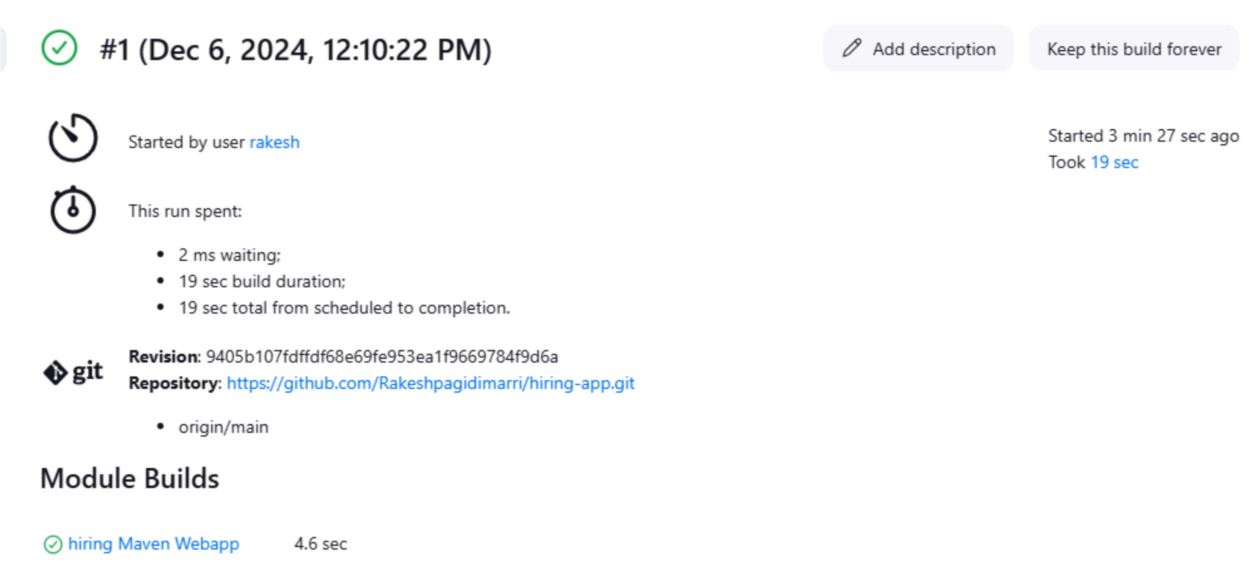
● Run regardless of build result

Should the post-build steps run only for successful builds, etc.

Invoke top-level Maven targets

Maven Version: MVN_HOME

Goals: ${MAVAN}



✓ #1 (Dec 6, 2024, 12:10:22 PM)

Started by user rakesh

Started 3 min 27 sec ago

Took 19 sec

This run spent:

- 2 ms waiting;
- 19 sec build duration;
- 19 sec total from scheduled to completion.

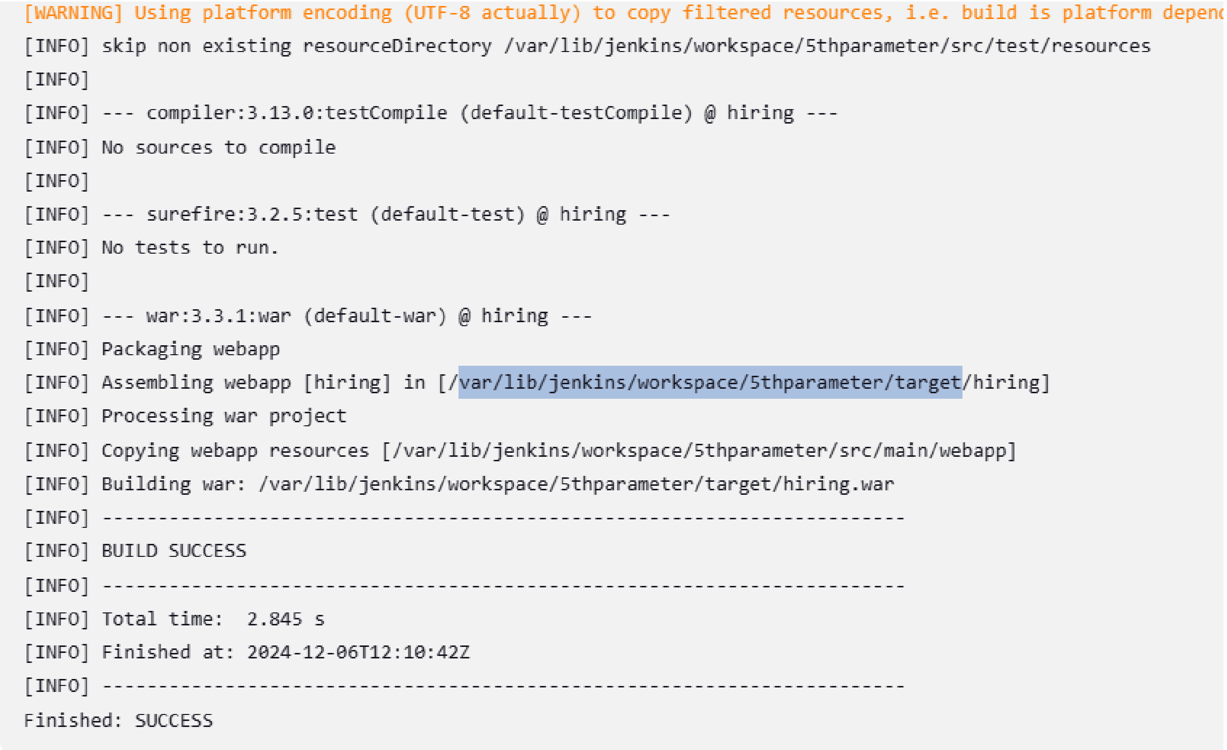
Revision: 9405b107fdffdf68e69fe953ea1f9669784f9d6a

Repository: https://github.com/Rakeshpagidimarri/hiring-app.git

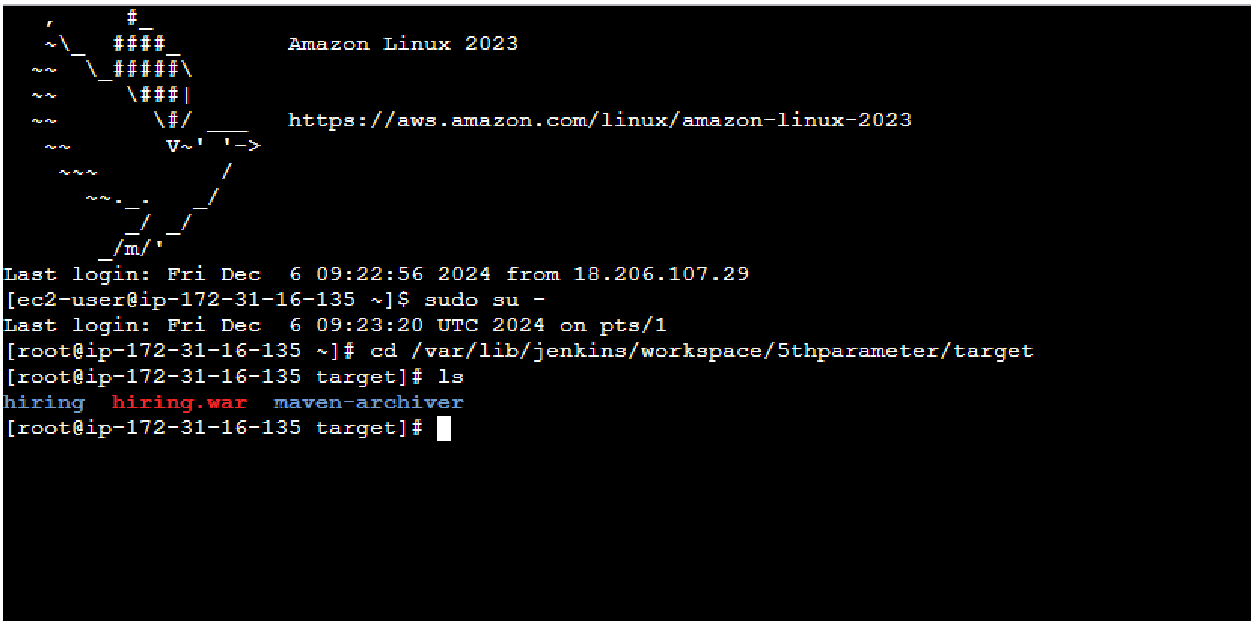
- origin/main

Module Builds

✓ hiring Maven Webapp 4.6 sec

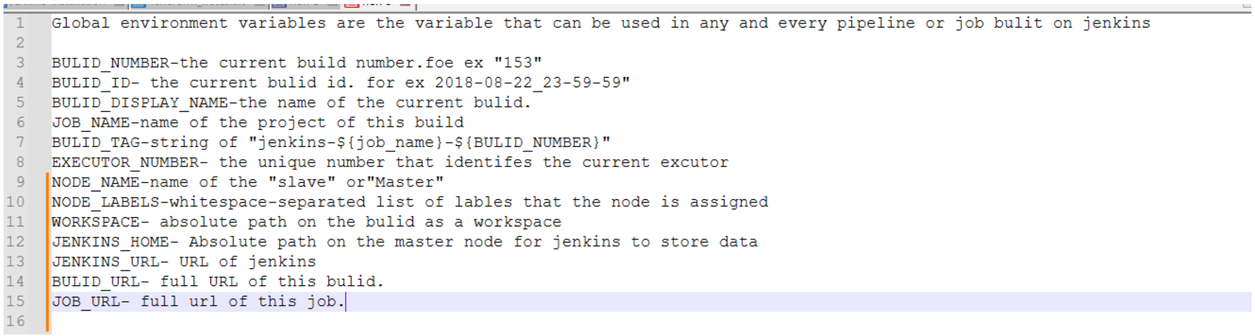


```
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform depend
[INFO] skip non existing resourceDirectory /var/lib/jenkins/workspace/5thparameter/src/test/resources
[INFO]
[INFO] --- compiler:3.13.0:testCompile (default-testCompile) @ hiring ---
[INFO] No sources to compile
[INFO]
[INFO] --- surefire:3.2.5:test (default-test) @ hiring ---
[INFO] No tests to run.
[INFO]
[INFO] --- war:3.3.1:war (default-war) @ hiring ---
[INFO] Packaging webapp
[INFO] Assembling webapp [hiring] in [/var/lib/jenkins/workspace/5thparameter/target/hiring]
[INFO] Processing war project
[INFO] Copying webapp resources [/var/lib/jenkins/workspace/5thparameter/src/main/webapp]
[INFO] Building war: /var/lib/jenkins/workspace/5thparameter/target/hiring.war
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  2.845 s
[INFO] Finished at: 2024-12-06T12:10:42Z
[INFO] ------------------------------------------------------------------------
Finished: SUCCESS
```



```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Dec  6 09:22:56 2024 from 18.206.107.29
[ec2-user@ip-172-31-16-135 ~]$ sudo su -
Last login: Fri Dec  6 09:23:20 UTC 2024 on pts/1
[root@ip-172-31-16-135 ~]# cd /var/lib/jenkins/workspace/5thparameter/target
[root@ip-172-31-16-135 target]# ls
hiring  hiring.war  maven-archiver
[root@ip-172-31-16-135 target]# 
```

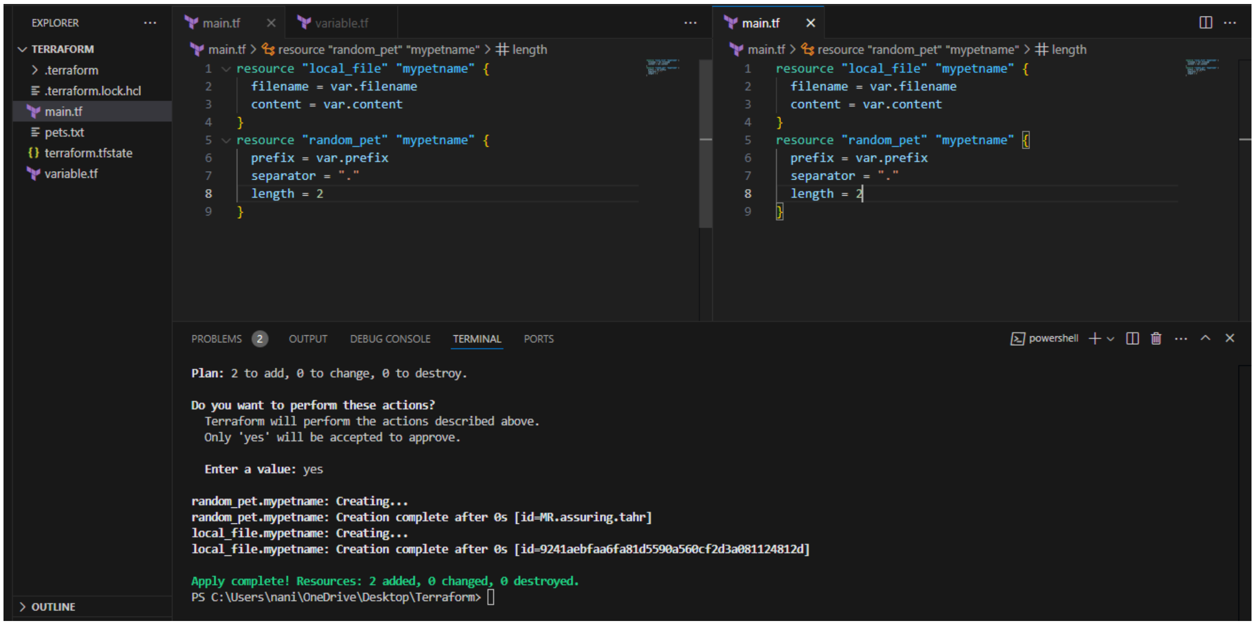
6) What are the global varaiables in jenkins?



```
Global environment variables are the variable that can be used in any and every pipeline or job bulit on jenkins

BULID_NUMBER-the current build number.foe ex "153"
BULID_ID- the current bulid id. for ex 2018-08-22_23-59-59"
BULID_DISPLAY_NAME-the name of the current bulid.
JOB_NAME-name of the project of this build
BULID_TAG-string of "jenkins-${job_name}-${BULID_NUMBER}"
EXECUTOR_NUMBER- the unique number that identifes the current excutor
NODE_NAME-name of the "slave" or"Master"
NODE_LABELS-whitespace-separated list of lables that the node is assigned
WORKSPACE- absolute path on the bulid as a workspace
JENKINS_HOME- Absolute path on the master node for jenkins to store data
JENKINS_URL- URL of jenkins
BULID_URL- full URL of this bulid.
JOB_URL- full url of this job.
```

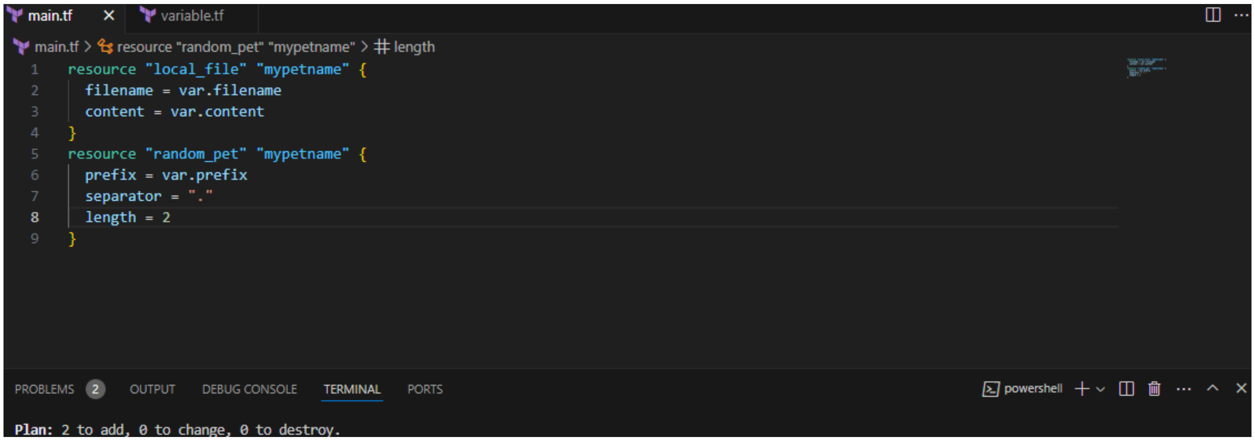
7) Watch terraform-04 video.

8) Execute the script shown in video.



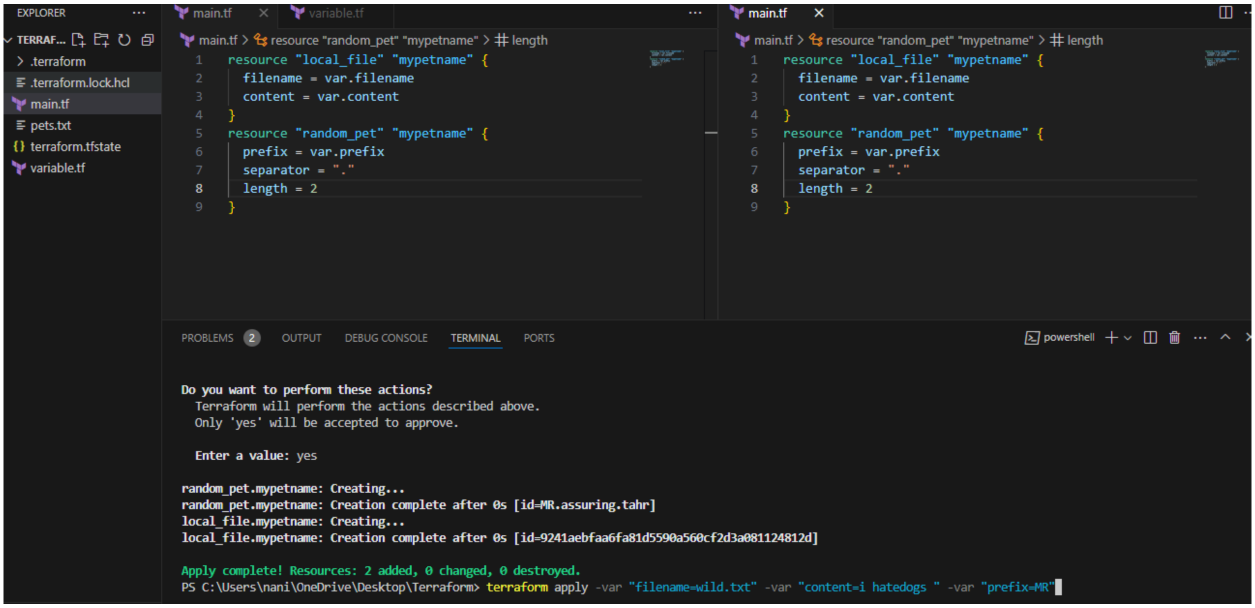
```
resource "local_file" "mypetname" {
  filename = var.filename
  content = var.content
}
resource "random_pet" "mypetname" {
  prefix = var.prefix
  separator = "."
  length = 2
}
```

```
Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

random_pet.mypetname: Creating...
random_pet.mypetname: Creation complete after 0s [id=MR.assuring.tahr]
local_file.mypetname: Creating...
local_file.mypetname: Creation complete after 0s [id=9241aebfaa6fa81d5590a560cf2d3a081124812d]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
PS C:\Users\nani\OneDrive\Desktop\Terraform>
```



```
resource "local_file" "mypetname" {
  filename = var.filename
  content = var.content
}
resource "random_pet" "mypetname" {
  prefix = var.prefix
  separator = "."
  length = 2
}
```

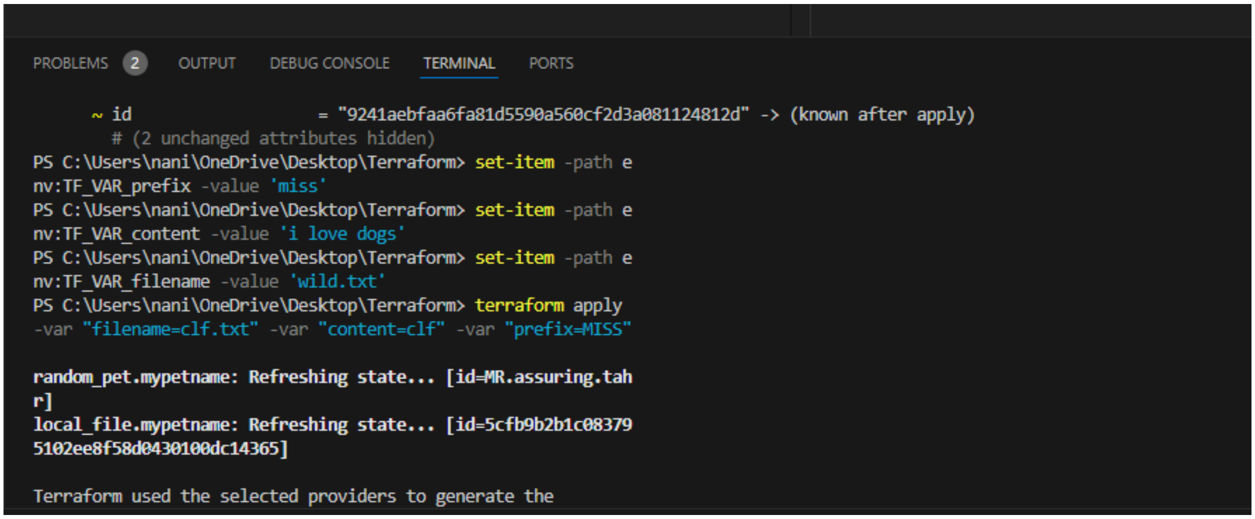
```
Plan: 2 to add, 0 to change, 0 to destroy.
```



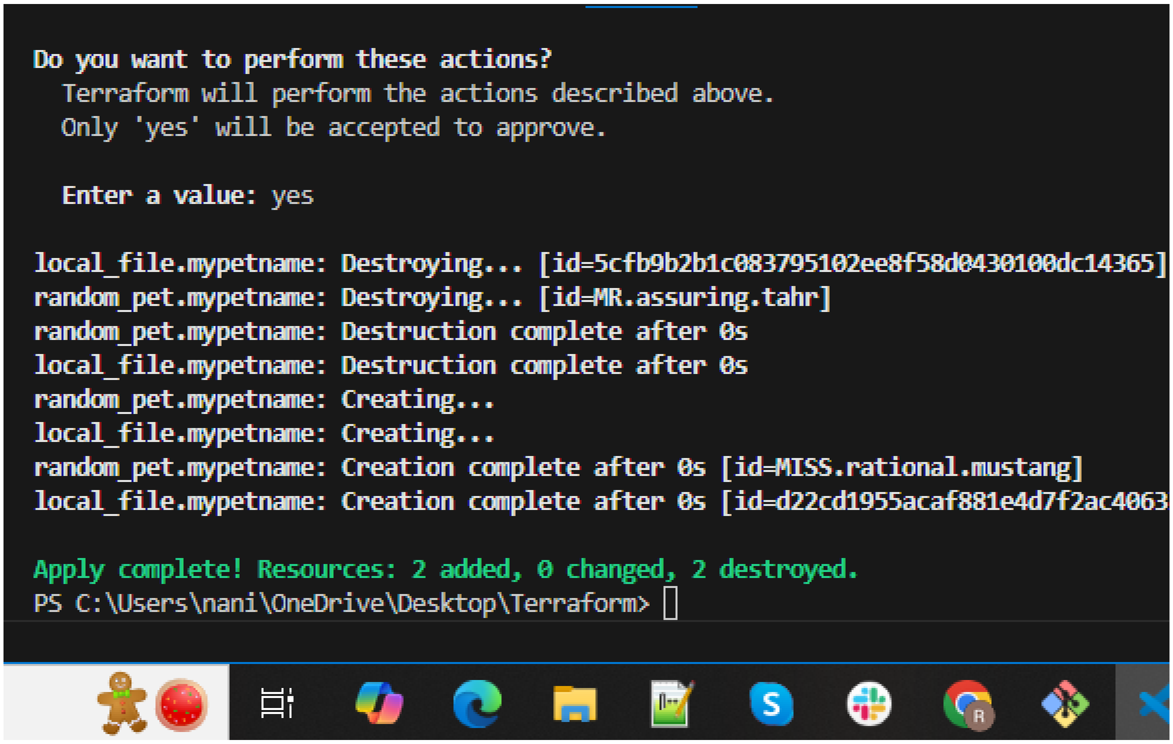
```
resource "local_file" "mypetname" {
  filename = var.filename
  content = var.content
}
resource "random_pet" "mypetname" {
  prefix = var.prefix
  separator = "."
  length = 2
}
```

```
resource "local_file" "mypetname" {
  filename = var.filename
  content = var.content
}
resource "random_pet" "mypetname" {
  prefix = var.prefix
  separator = "."
  length = 2
}
```

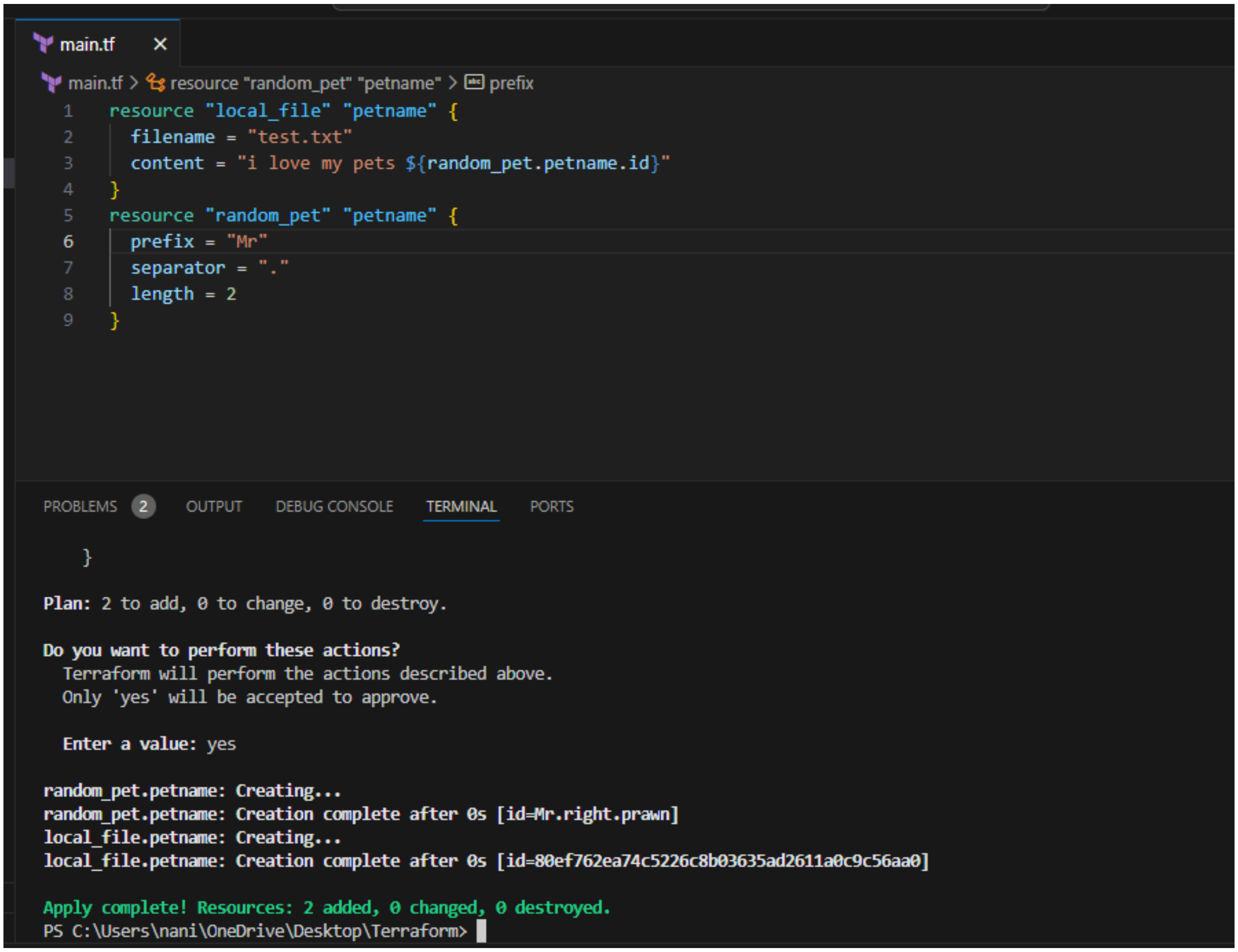
```
Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

random_pet.mypetname: Creating...
random_pet.mypetname: Creation complete after 0s [id=MR.assuring.tahr]
local_file.mypetname: Creating...
local_file.mypetname: Creation complete after 0s [id=9241aebfaa6fa81d5590a560cf2d3a081124812d]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
PS C:\Users\nani\OneDrive\Desktop\Terraform> terraform apply -var "filename=wild.txt" -var "content=i hatedogs " -var "prefix=MR"
```



```
      ~ id                   = "9241aebfaa6fa81d5590a560cf2d3a081124812d" -> (known after apply)
        # (2 unchanged attributes hidden)
PS C:\Users\nani\OneDrive\Desktop\Terraform> set-item -path e
nv:TF_VAR_prefix -value 'miss'
PS C:\Users\nani\OneDrive\Desktop\Terraform> set-item -path e
nv:TF_VAR_content -value 'i love dogs'
PS C:\Users\nani\OneDrive\Desktop\Terraform> set-item -path e
nv:TF_VAR_filename -value 'wild.txt'
PS C:\Users\nani\OneDrive\Desktop\Terraform> terraform apply
-var "filename=clf.txt" -var "content=clf" -var "prefix=MISS"

random_pet.mypetname: Refreshing state... [id=MR.assuring.tah
r]
local_file.mypetname: Refreshing state... [id=5cfb9b2b1c08379
5102ee8f58d0430100dc14365]

Terraform used the selected providers to generate the
```



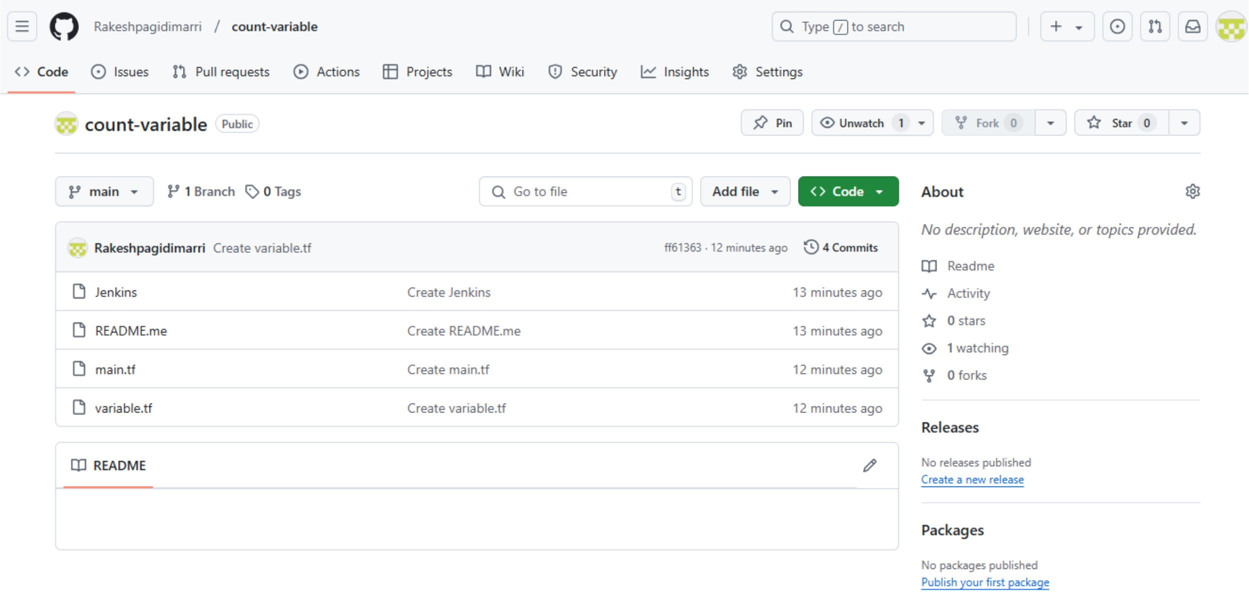
```
Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

local_file.mypetname: Destroying... [id=5cfb9b2b1c083795102ee8f58d0430100dc14365]
random_pet.mypetname: Destroying... [id=MR.assuring.tahr]
random_pet.mypetname: Destruction complete after 0s
local_file.mypetname: Destruction complete after 0s
random_pet.mypetname: Creating...
local_file.mypetname: Creating...
random_pet.mypetname: Creation complete after 0s [id=MISS.rational.mustang]
local_file.mypetname: Creation complete after 0s [id=d22cd1955acaf881e4d7f2ac4063

Apply complete! Resources: 2 added, 0 changed, 2 destroyed.
PS C:\Users\nani\OneDrive\Desktop\Terraform>
```



```
resource "local_file" "petname" {
  filename = "test.txt"
  content = "i love my pets ${random_pet.petname.id}"
}
resource "random_pet" "petname" {
  prefix = "Mr"
  separator = "."
  length = 2
}
```

```
    }

Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

random_pet.petname: Creating...
random_pet.petname: Creation complete after 0s [id=Mr.right.prawn]
local_file.petname: Creating...
local_file.petname: Creation complete after 0s [id=80ef762ea74c5226c8b03635ad2611a0c9c56aa0]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
PS C:\Users\nani\OneDrive\Desktop\Terraform>
```

9) Integrate terrafrom in jenkins using Terraform plugin.



Rakeshpagidimarri Create variable.tf ff61363 · 12 minutes ago 4 Commits

Jenkins	Create Jenkins	13 minutes ago
README.me	Create README.me	13 minutes ago
main.tf	Create main.tf	12 minutes ago
variable.tf	Create variable.tf	12 minutes ago

README

About

No description, website, or topics provided.

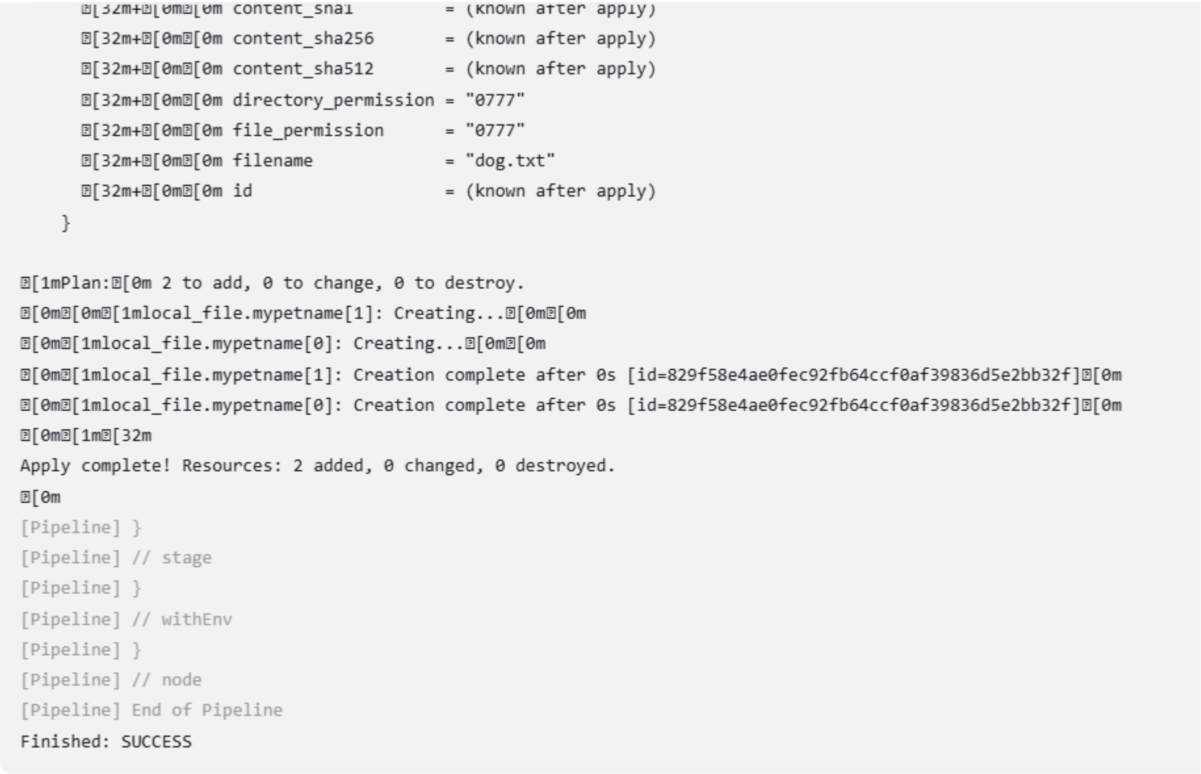
Readme
Activity
0 stars
1 watching
0 forks

Releases

No releases published
Create a new release

Packages

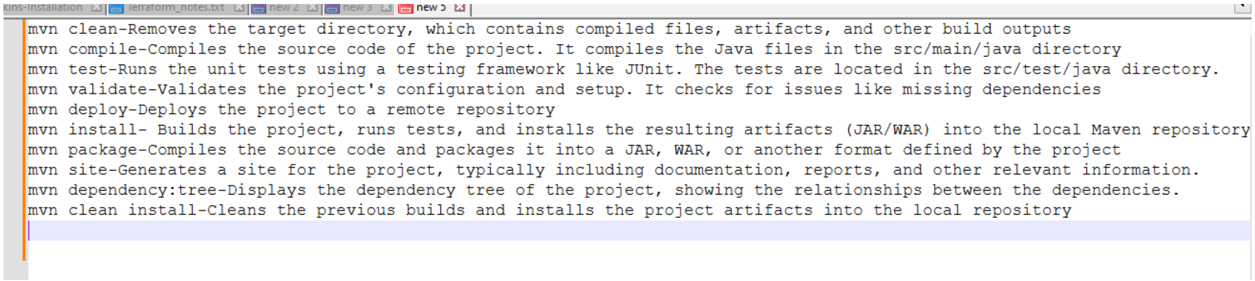
No packages published
Publish your first package



```
      □[32m+□[0m□[0m content_sha1          = (known after apply)
      □[32m+□[0m□[0m content_sha256        = (known after apply)
      □[32m+□[0m□[0m content_sha512        = (known after apply)
      □[32m+□[0m□[0m directory_permission = "0777"
      □[32m+□[0m□[0m file_permission      = "0777"
      □[32m+□[0m□[0m filename             = "dog.txt"
      □[32m+□[0m□[0m id                   = (known after apply)
    }

□[1mPlan:□[0m 2 to add, 0 to change, 0 to destroy.
□[0m□[0m□[1mlocal_file.mypetname[1]: Creating...□[0m□[0m
□[0m□[1mlocal_file.mypetname[0]: Creating...□[0m□[0m
□[0m□[1mlocal_file.mypetname[1]: Creation complete after 0s [id=829f58e4ae0fec92fb64ccf0af39836d5e2bb32f]□[0m
□[0m□[1mlocal_file.mypetname[0]: Creation complete after 0s [id=829f58e4ae0fec92fb64ccf0af39836d5e2bb32f]□[0m
□[0m□[1m□[32m
Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
□[0m
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

10) Create CICD pipeline for Nodejs Application.
https://github.com/betawins/Trading-UI.git

11) Explain 10 Maven commands.



```
mvn clean-Removes the target directory, which contains compiled files, artifacts, and other build outputs
mvn compile-Compiles the source code of the project. It compiles the Java files in the src/main/java directory
mvn test-Runs the unit tests using a testing framework like JUnit. The tests are located in the src/test/java directory.
mvn validate-Validates the project's configuration and setup. It checks for issues like missing dependencies
mvn deploy-Deploys the project to a remote repository
mvn install- Builds the project, runs tests, and installs the resulting artifacts (JAR/WAR) into the local Maven repository
mvn package-Compiles the source code and packages it into a JAR, WAR, or another format defined by the project
mvn site-Generates a site for the project, typically including documentation, reports, and other relevant information.
mvn dependency:tree-Displays the dependency tree of the project, showing the relationships between the dependencies.
mvn clean install-Cleans the previous builds and installs the project artifacts into the local repository
```