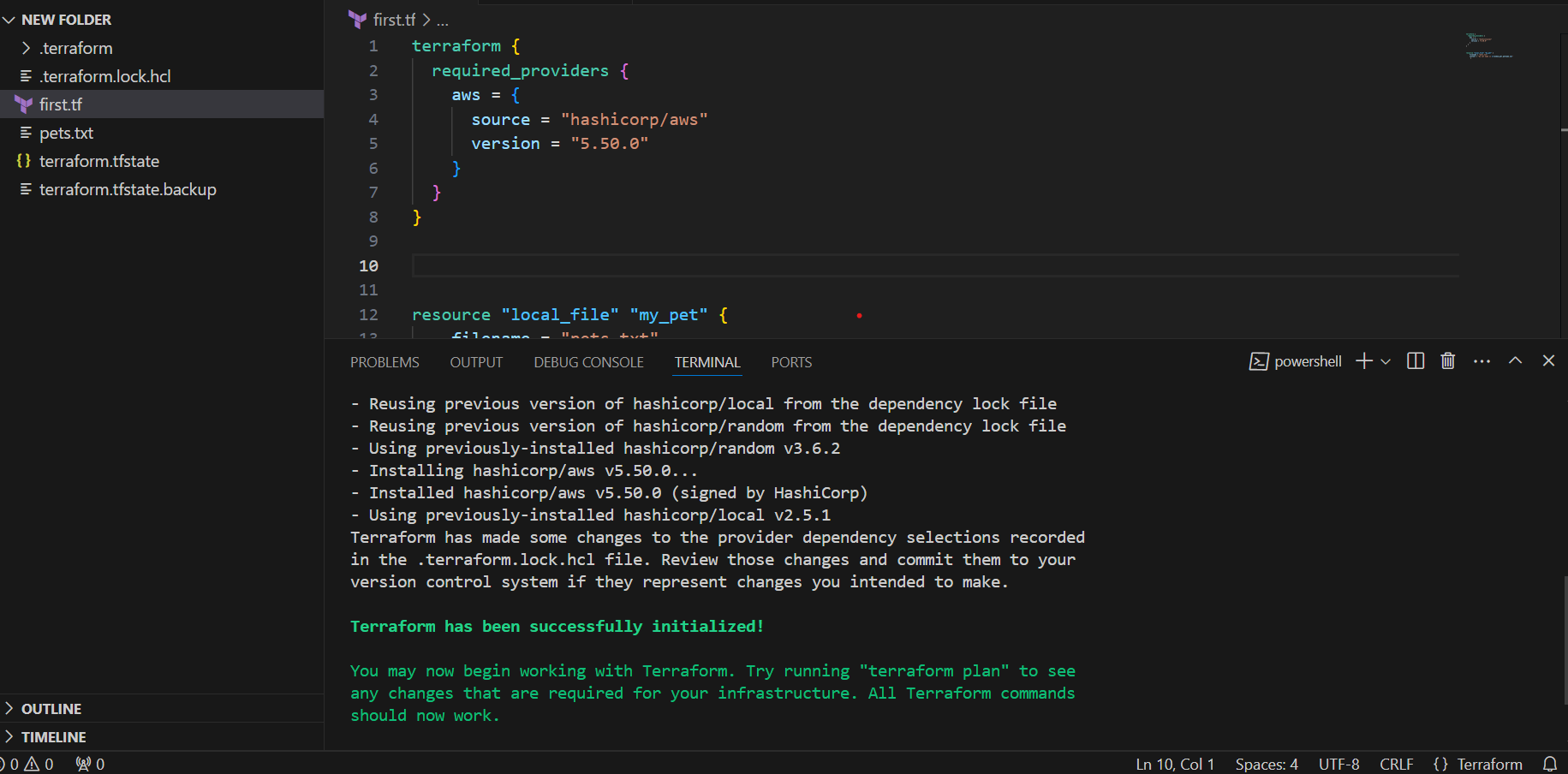
1. Watch terraform-04 video.

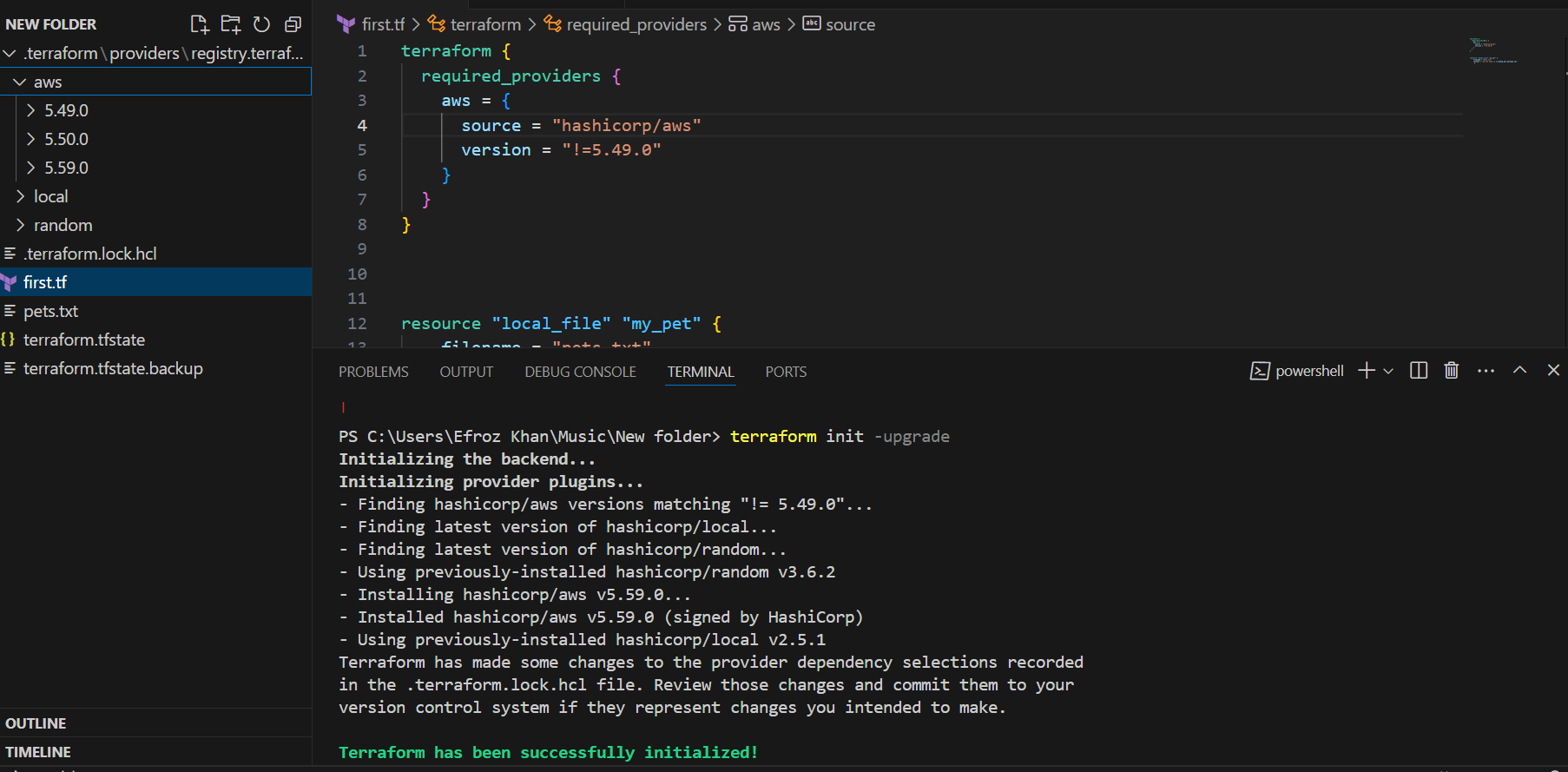
**Watched**

1. Execute the script shown in video.

Changing the provider version constants



Changing the version !=”5.50.0” except this version uses another version.



**=========**

**version = "2.3.0" --> download the exact version**

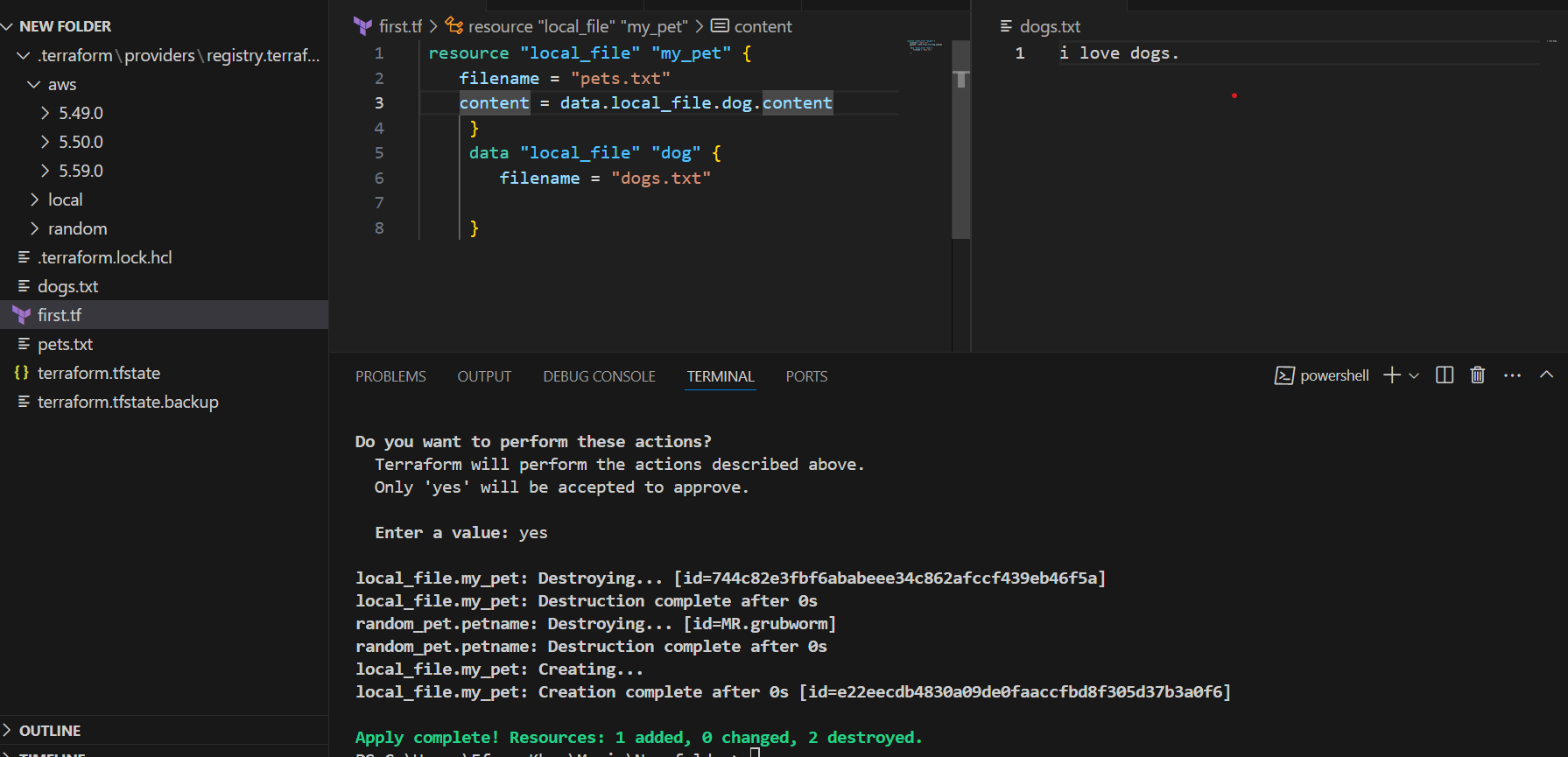
**version = "!=2.3.0" --> will not use the mentioned version**

**version = "< 2.3.0" --> lesses than the mention version**

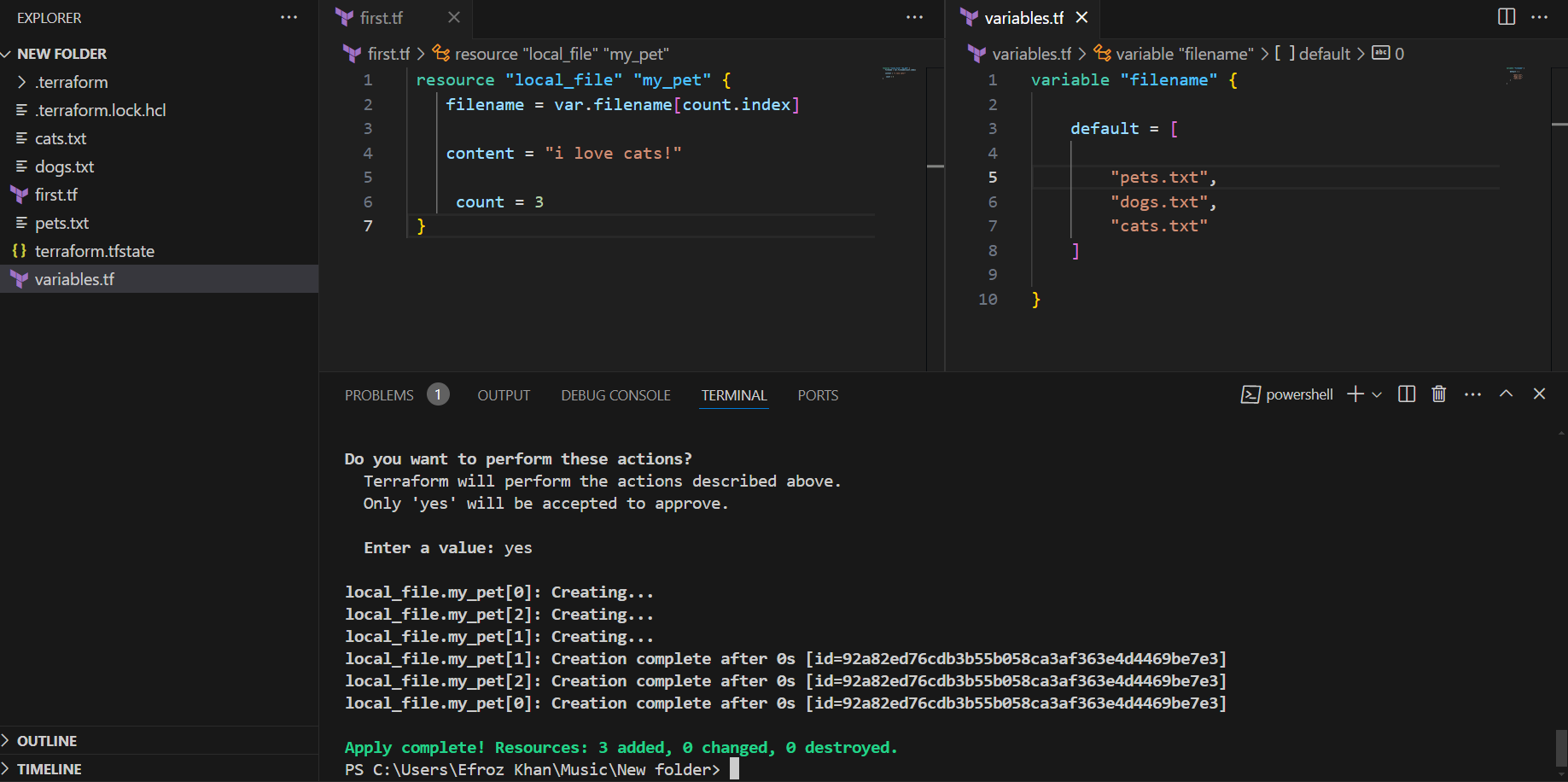
**version = "> 2.3.0" --> greater than the given version**

**version = "~> 2.3.0" --> specific version or higher version.**

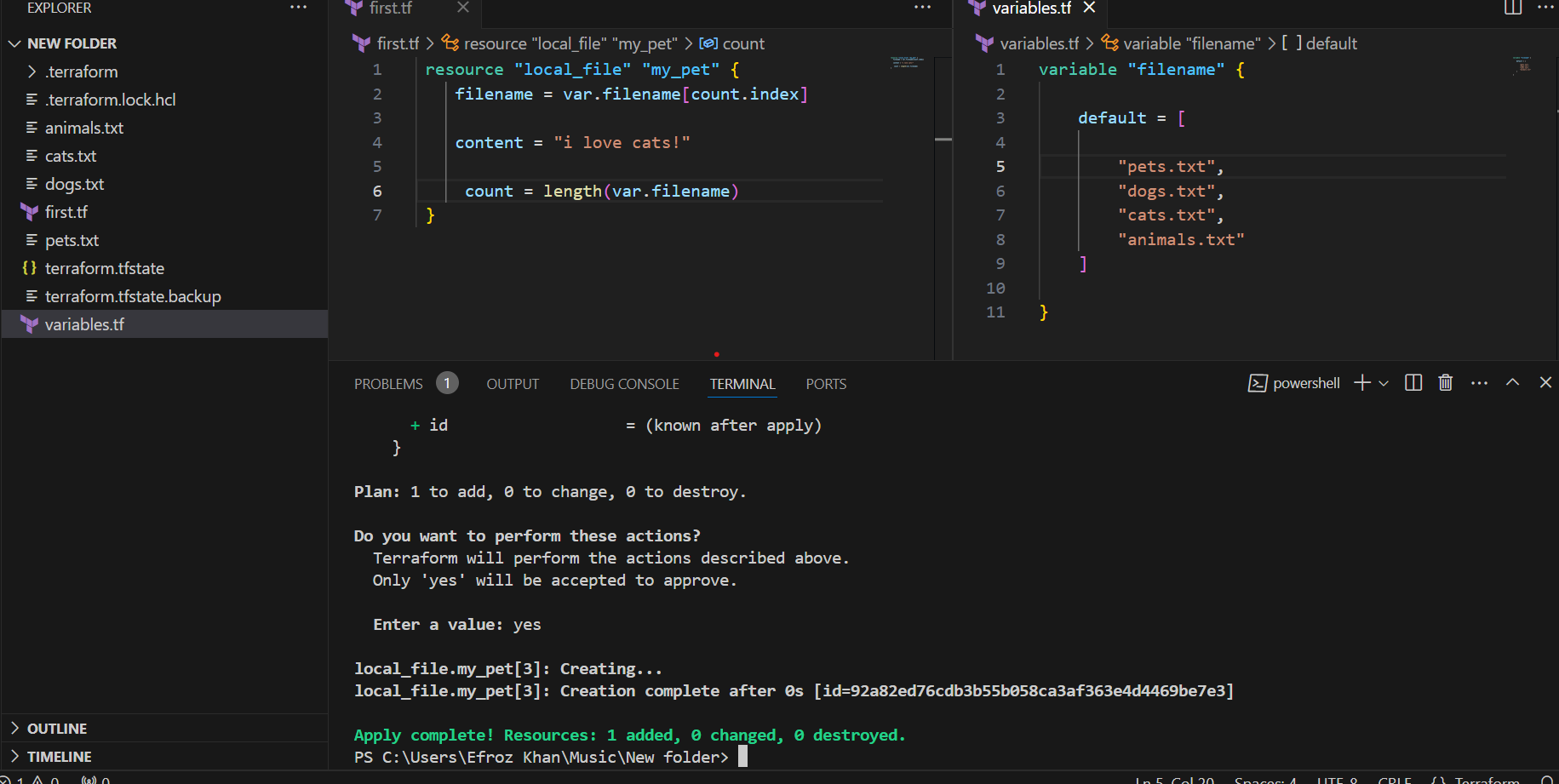
**Data blog is used to read content of file**

****

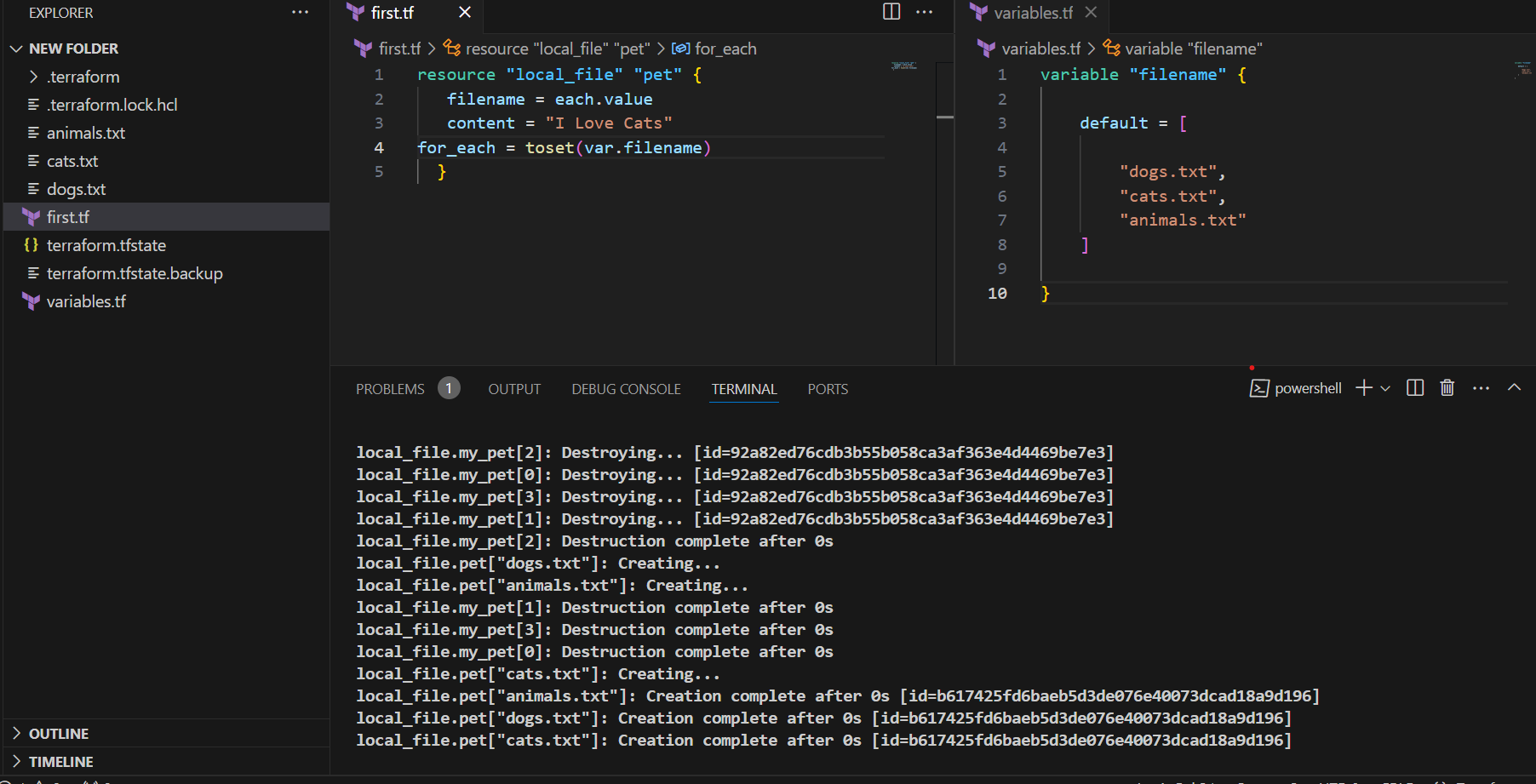
**Meta Arguments (Counts)**

****

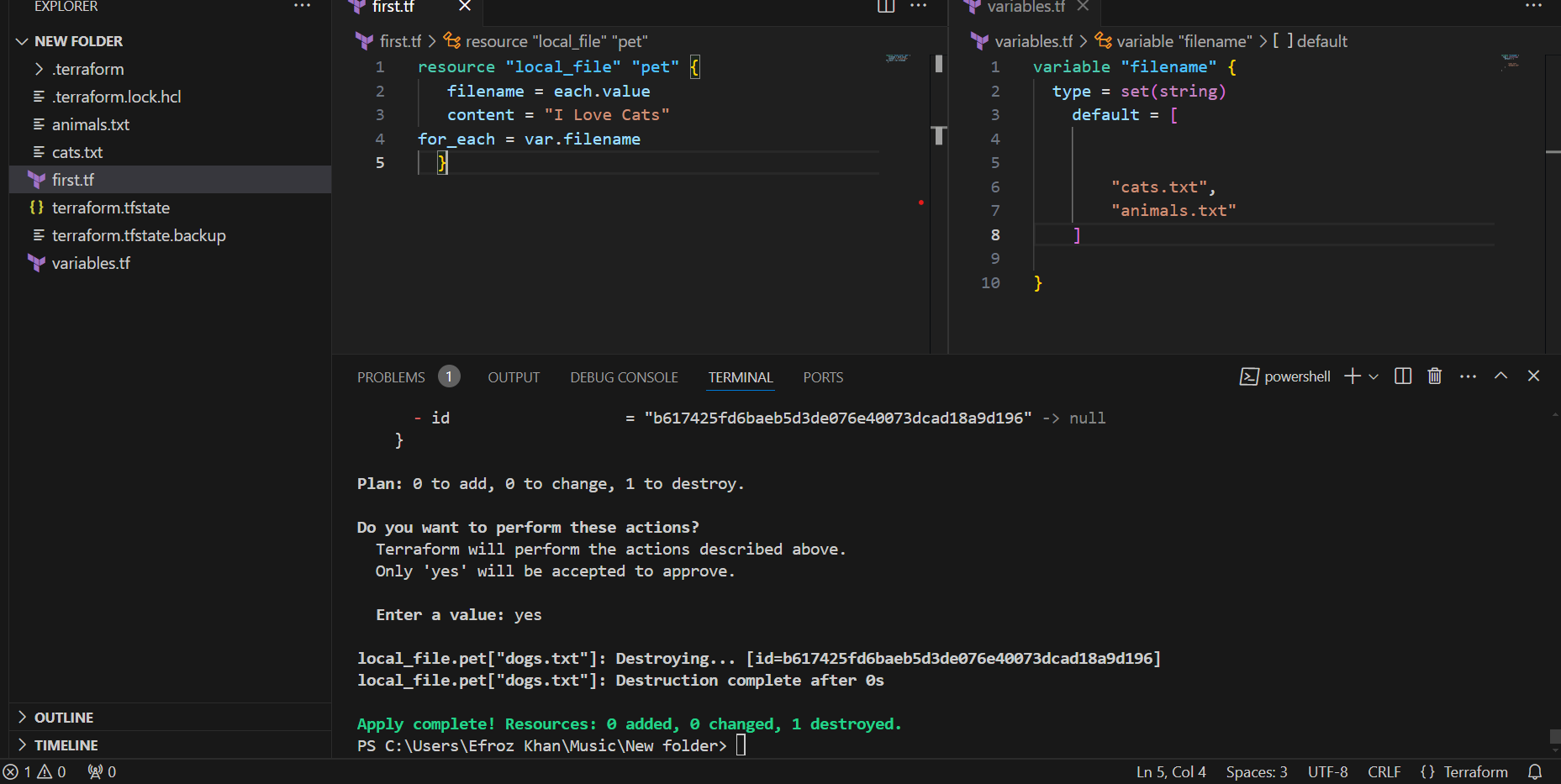
**Count (length function)=length(var.filename)**

****

**Argument (for\_each) :-**

****

**Meta arguments:- If we don’t want to use in built function then go for type which in varable type = set(string)**

****

**Creating iam user using terraform resource:-**

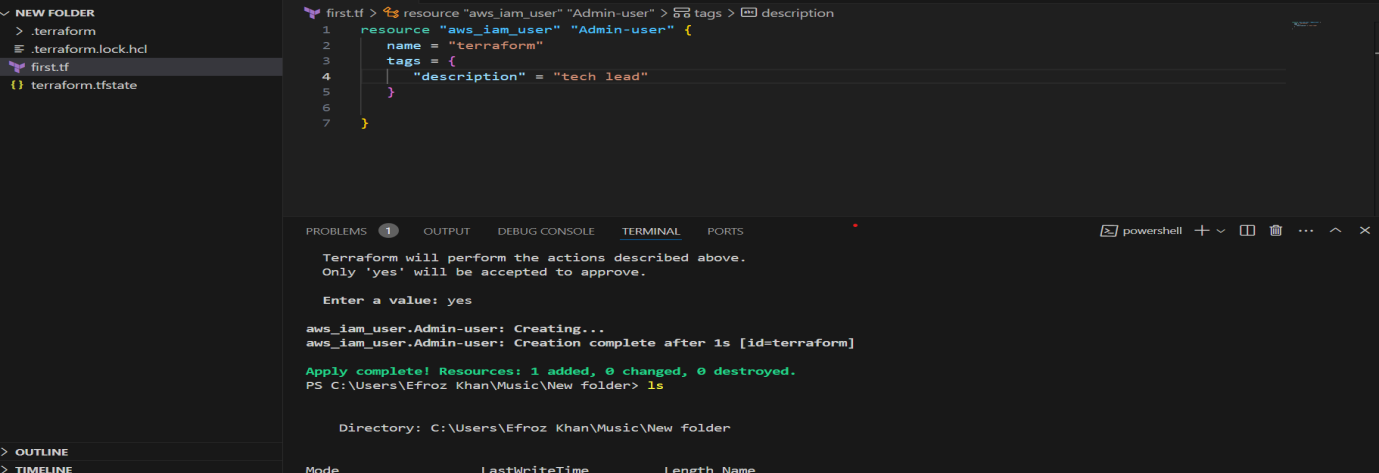
**First configure aws credentials in locally**

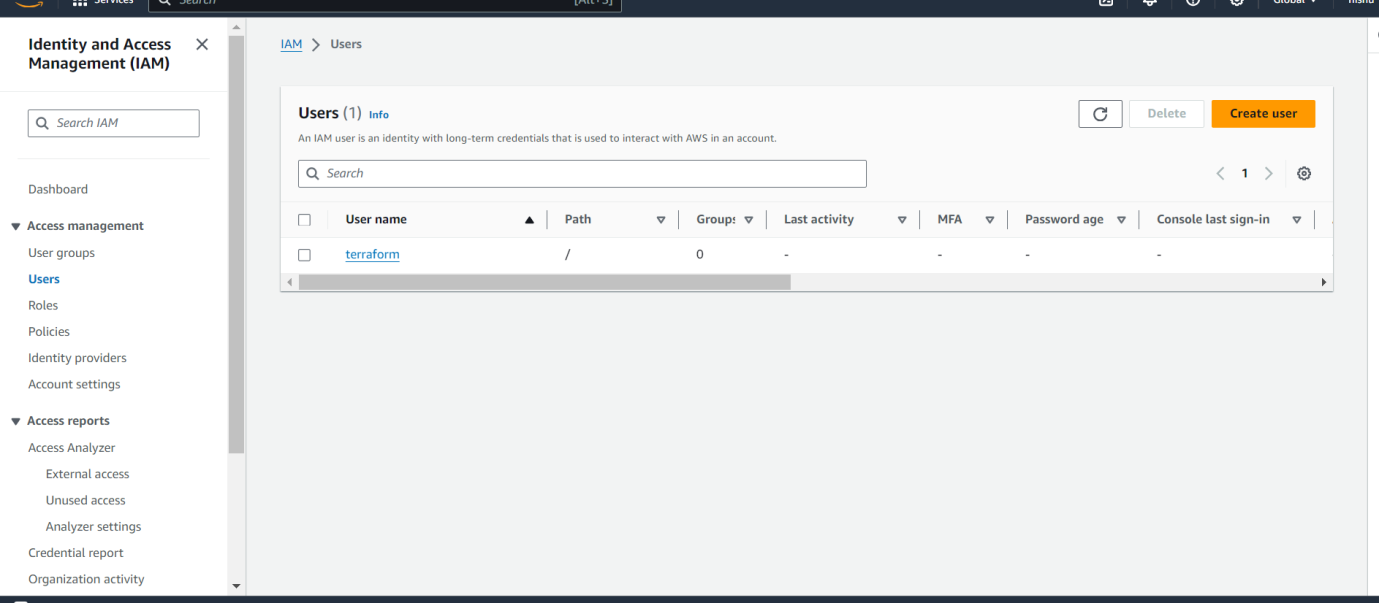
**Once crateing he resource file for creating user**

**Then save use this commands**

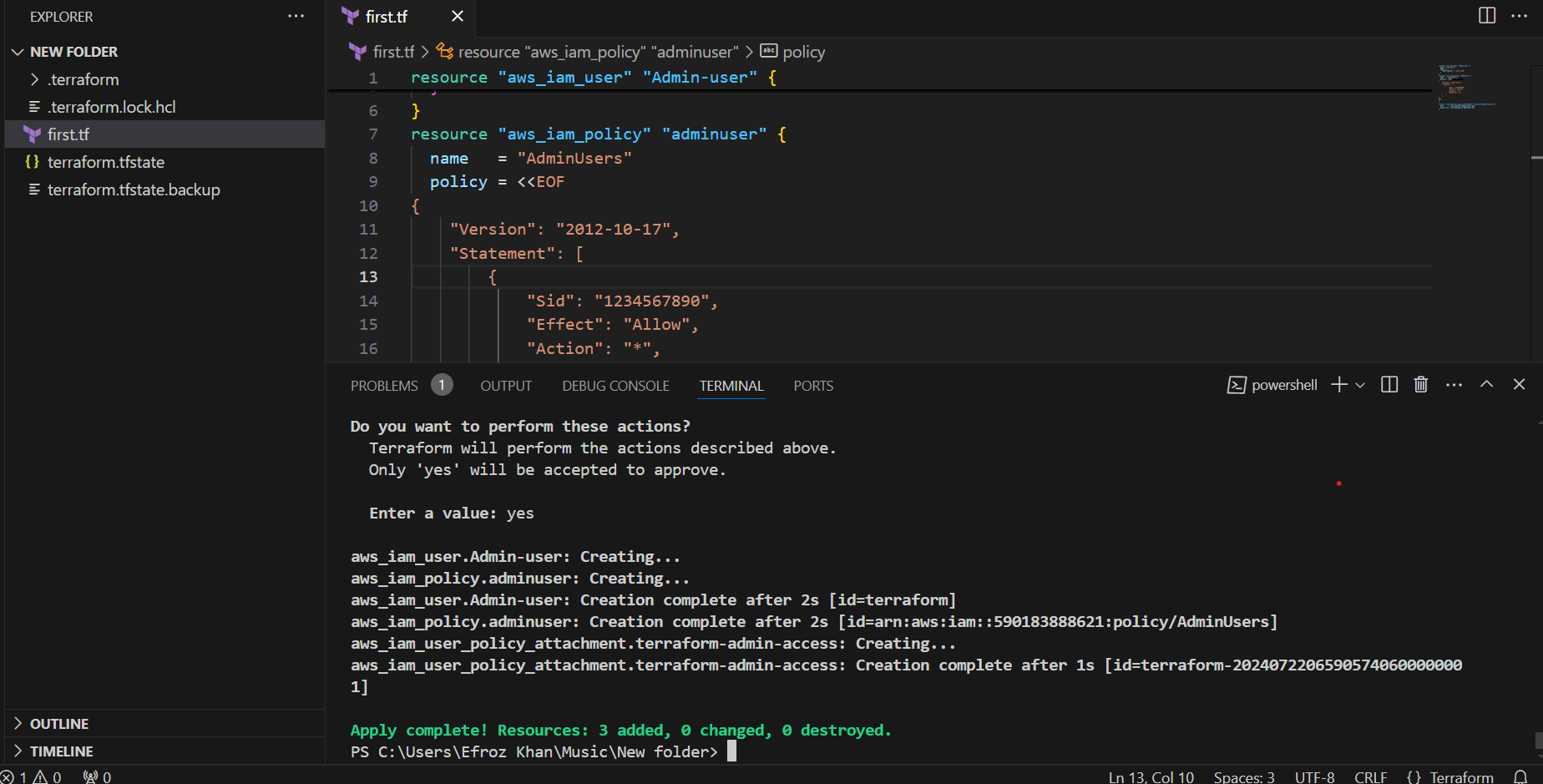
**Terraform init (use for installing related aws user plugins)**

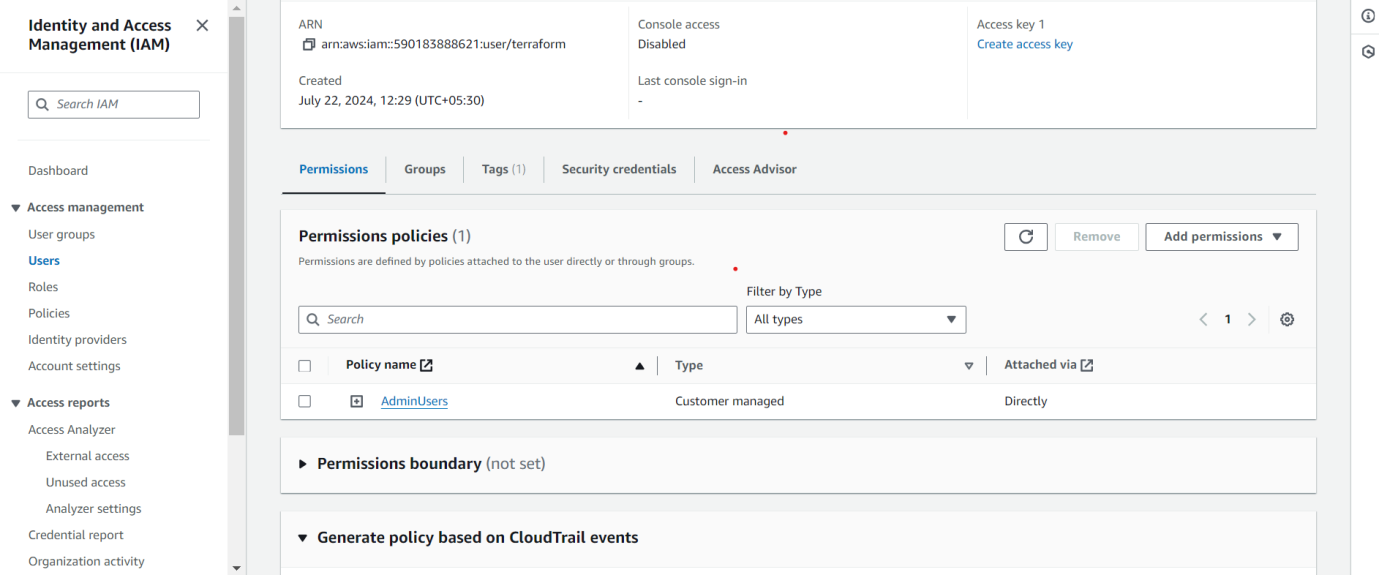
**Terraform apply (its create the user go and check aws console in iam)**

****

****

**Here doc syntax:-**

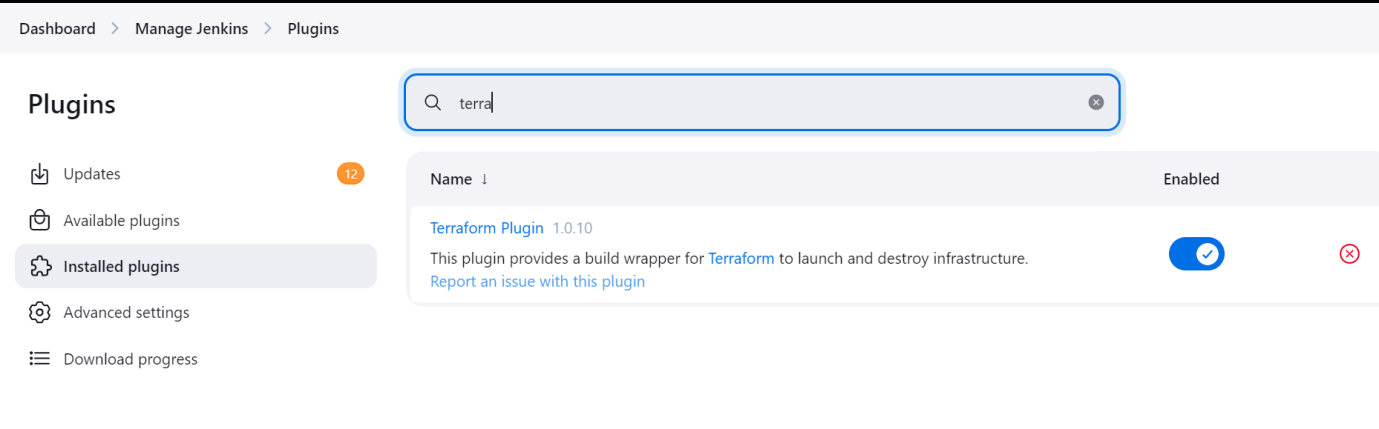
****

****

**Creating user terraform and attaching the admin-user policy using three resouces**

1. Integrate terrafrom in jenkins using Terraform plugin.

Install terraform plugin in Jenkins server.



Then install terraform in Jenkins server

**Package manager:**

sudo yum install -y yum-utils shadow-utils

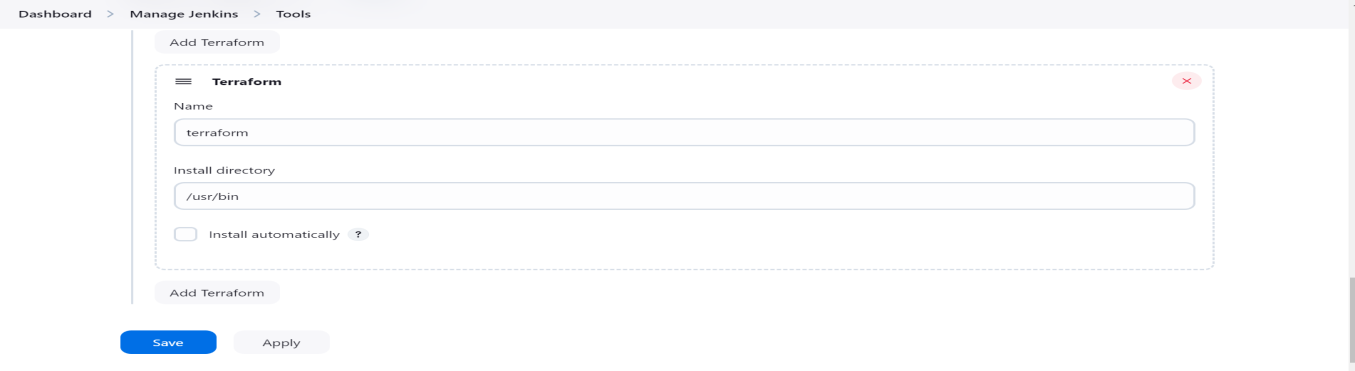
sudo yum-config-manager --add-repo https://rpm.releases.hashicorp.com/AmazonLinux/hashicorp.repo

sudo yum -y install terraform

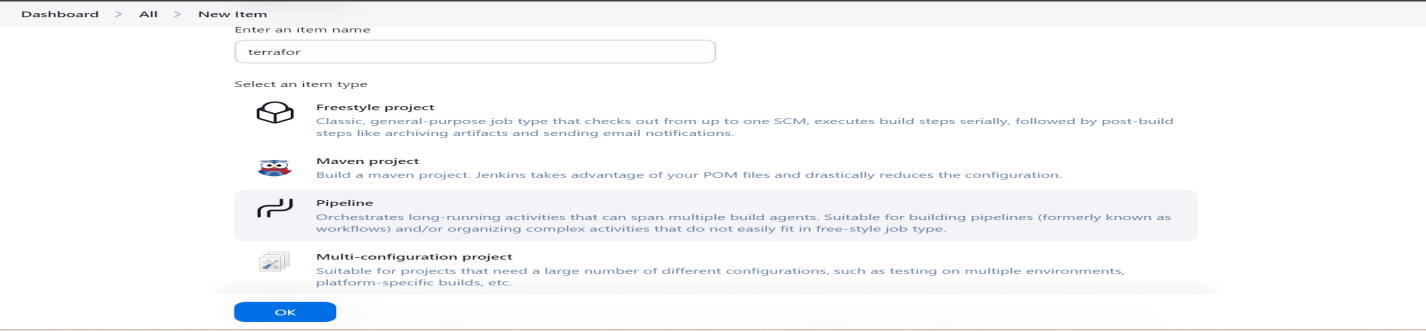
once done with installation check version using the command

Mention terrform path in the global tool configuration.

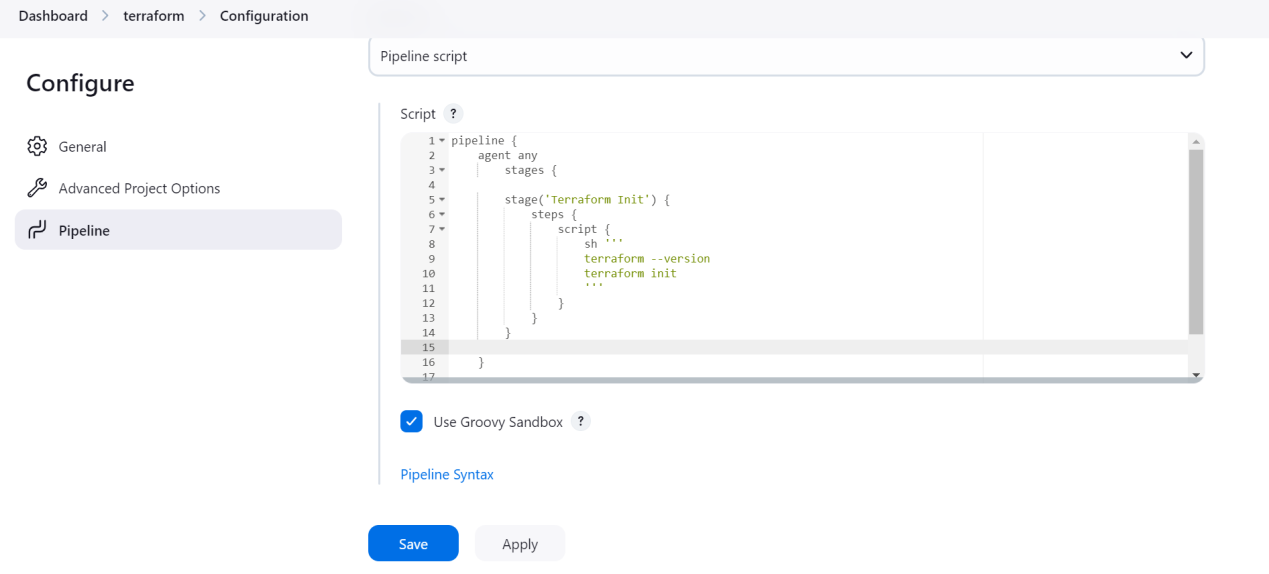
terraform --vesion

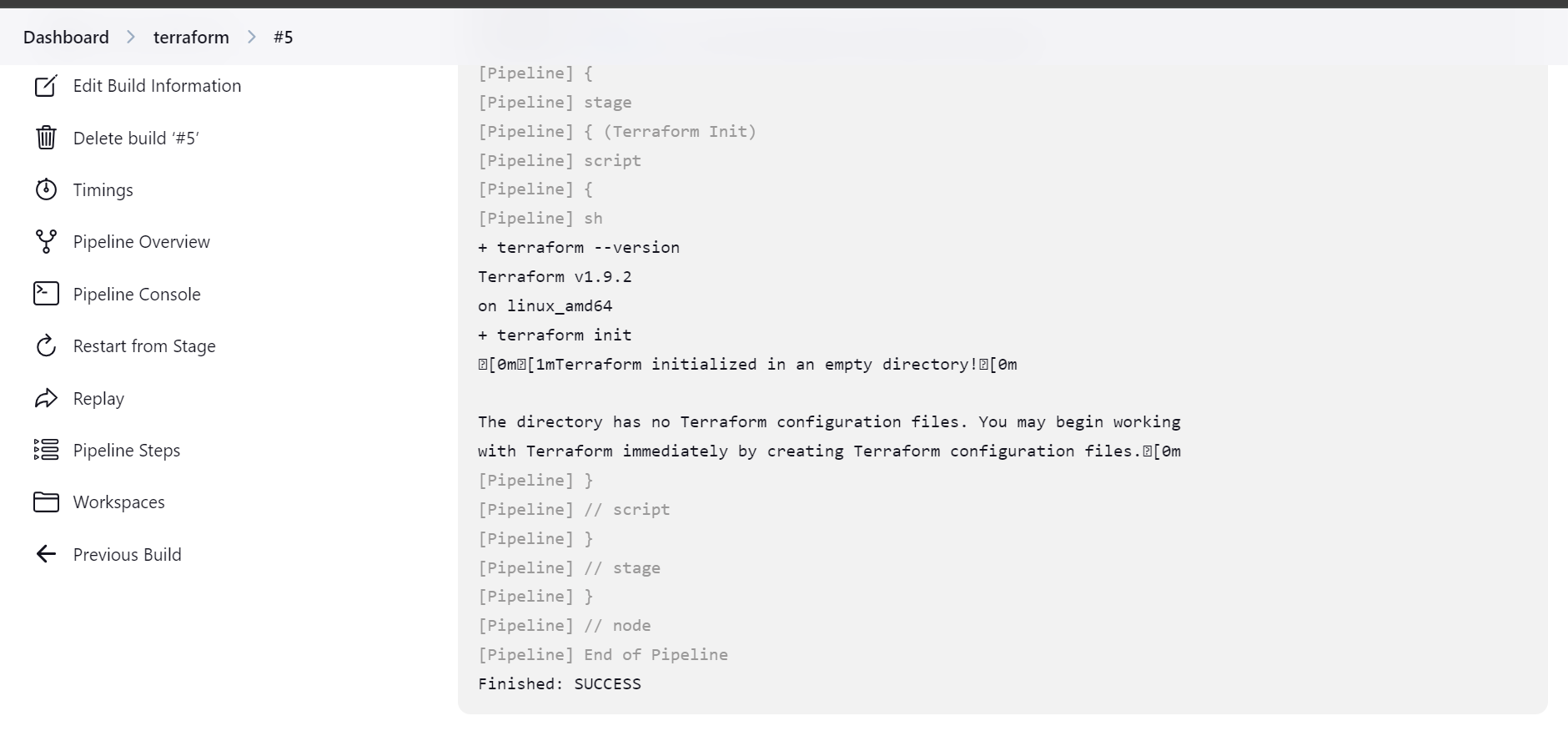


then create the Jenkins pipeline job

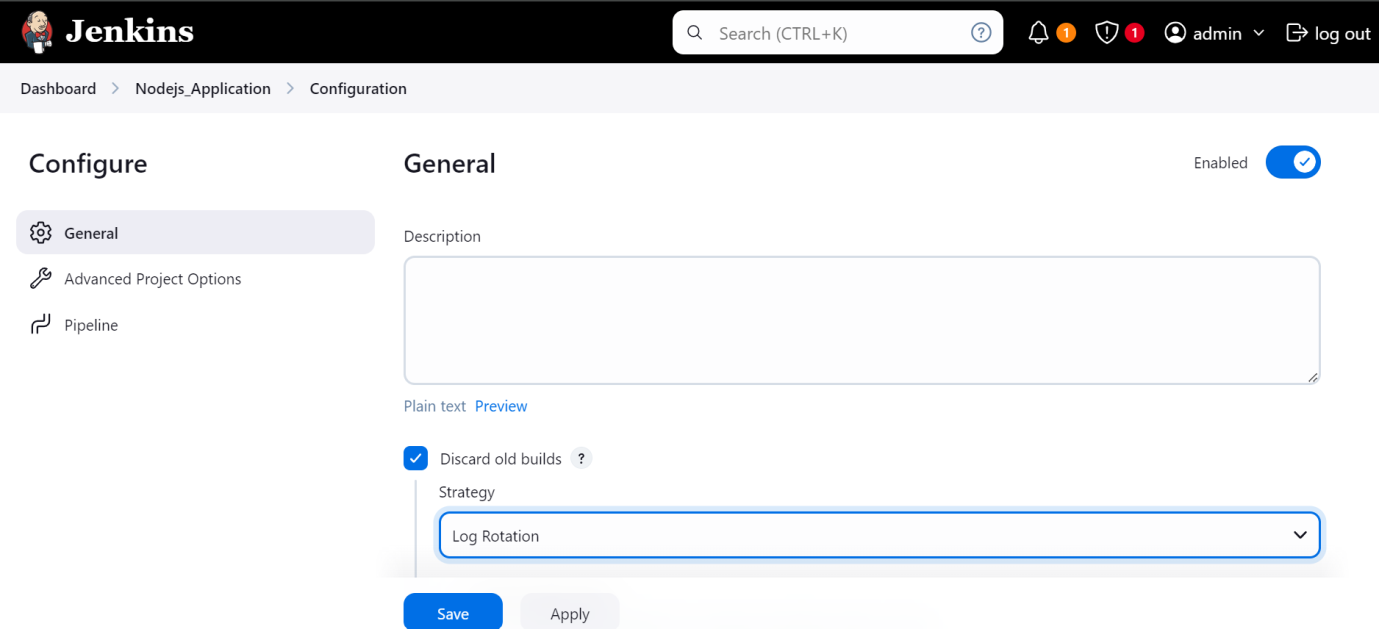


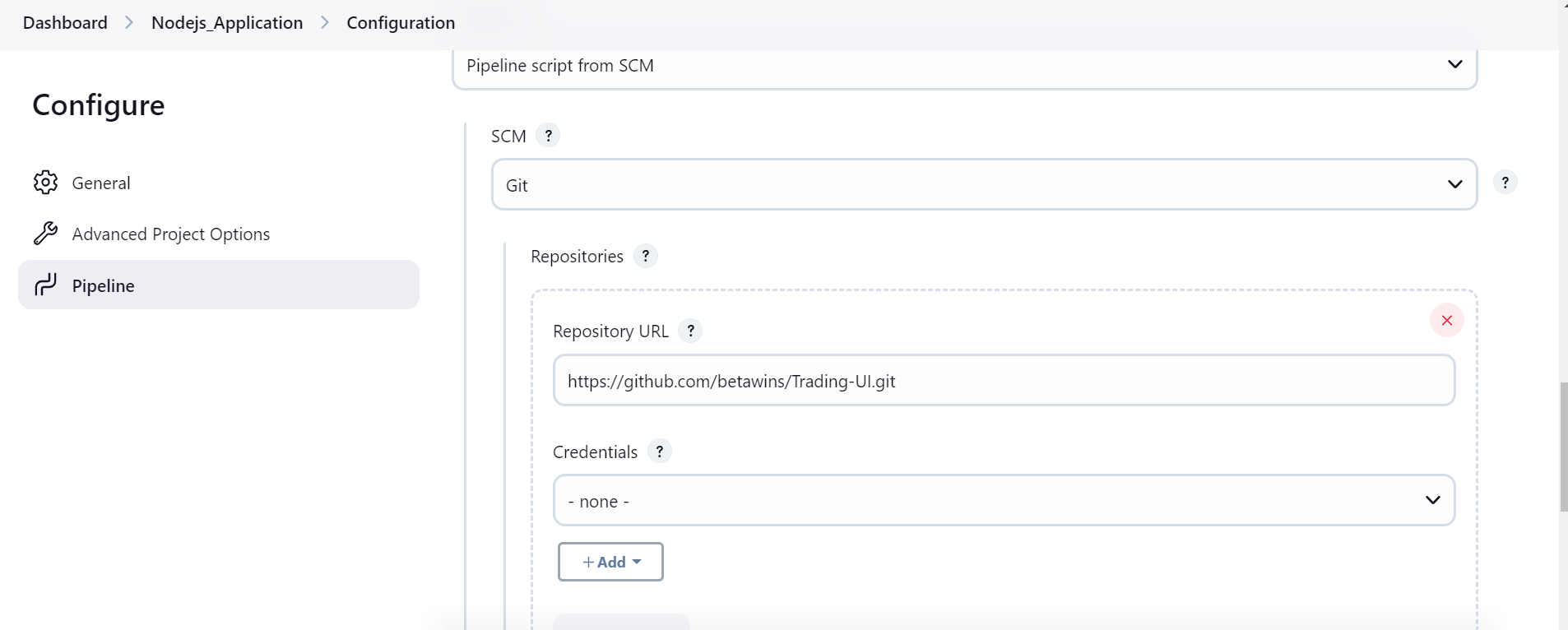
Write pipeline script

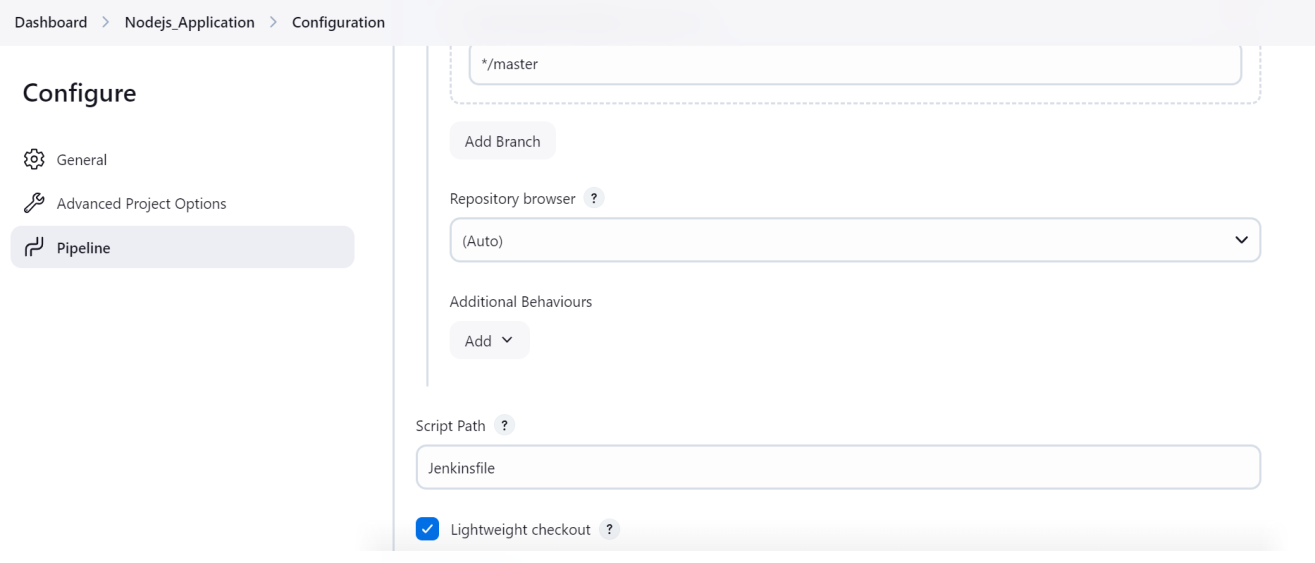


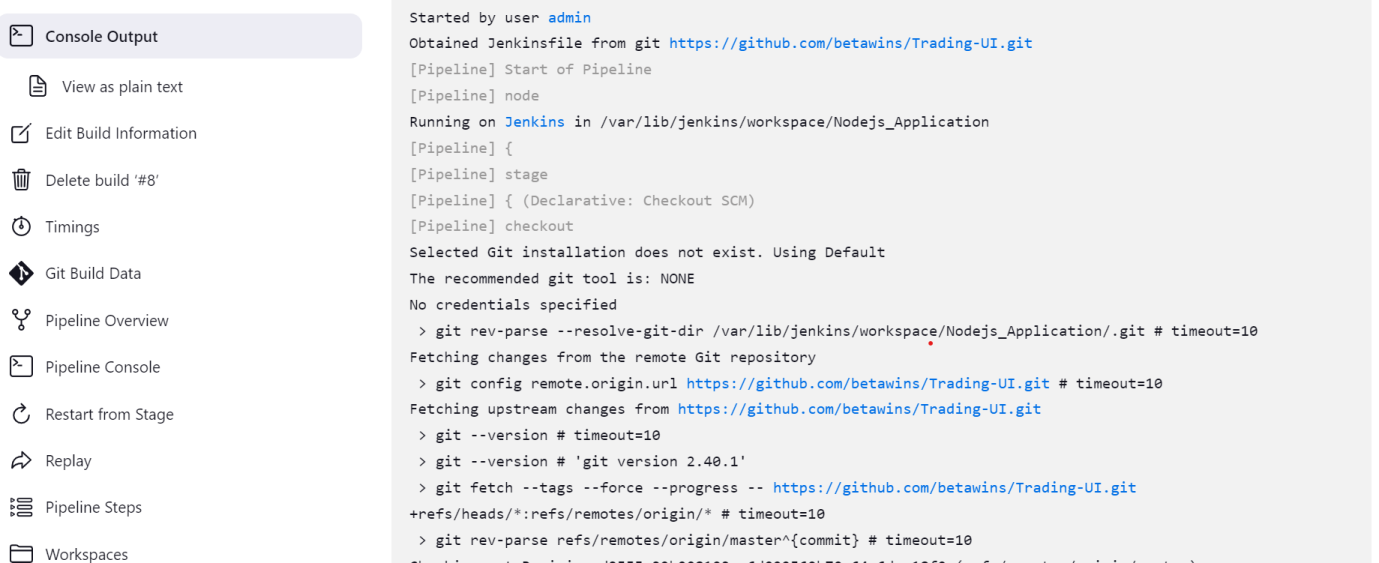


1. Create CICD pipeline for Nodejs Application. <https://github.com/betawins/Trading-UI.git>









5) Explain 10 Maven commands.

1. **mvn clean:** Cleans the project by deleting the target directory which contains compiled   
classes, generated JARs, and other build artifacts.

2. **mvn compile:** Compiles the source code of the project.  
3. **mvn test:** Runs the unit tests for the project.  
4. **mvn package:** Packages the compiled code into a distributable format, such as a JAR or WAR  
file.  
5. **mvn install:** Installs the packaged artifact into the local Maven repository, making it available   
for other projects locally.  
6. **mvn deploy:** Deploys the packaged artifact to a remote repository for sharing with other   
developers or projects.  
7.**mvn dependency:** Displays the dependency tree for the project, showing all dependencies and their transitive dependencies.  
8.**mvn dependency:** Resolves and downloads all dependencies required for the project.  
9. **mvn site:** Generates project documentation, including reports, and places them in the target/site directory.  
10.**mvn hel:** Displays the effective POM (Project Object Model) after inheritance and interpolation,   
showing the final configuration of the project.